FUJITSU SEMICONDUCTOR
CONTROLLER MANUAL

CM25-00323-5E

F²MC-8L/8FX FAMILY

SOFTUNE™ Workbench

COMMAND REFERENCE MANUAL

FUJITSU

F²MC-8L/8FX FAMILY

SOFTUNE™ Workbench

COMMAND REFERENCE MANUAL

FUJITSU SEMICONDUCTOR LIMITED

PREFACE

■ What is the SOFTUNE Workbench?

SOFTUNE Workbench is support software for developing programs for the F^2MC-8L/8FX families of microcontrollers.

It is a combination of a development manager, simulator debugger, emulator debugger, monitor debugger, and an integrated development environment for efficient development.

Note: F^2MC is the abbreviation of FUJITSU Flexible Microcontroller.

■ Purpose of this manual and target readers

This manual explains the command and built-in variable/function of the SOFTUNE Workbench in the reference format.

This manual is intended for engineers developing various types of products using SOFTUNE Workbench.

Be sure to read this manual completely.

■ Trademarks

SOFTUNE is a trademark of FUJITSU SEMICONDUCTOR LIMITED.

Windows is registered trademarks of Microsoft Corporation in the U.S. and other countries.

The company names and brand names herein are the trademarks or registered trademarks of their respective owners.

■ Organization of Manual

This manual consists of 14 chapters and an appendix.

CHAPTER 1 Environment Setup Commands

This chapter describes the Environment Setup commands.

CHAPTER 2 Program Execution Commands

This chapter describes the Program Execution commands.

CHAPTER 3 Break/Event Control Command

This chapter describes the Break/Event Control commands.

CHAPTER 4 Program Execution Analysis Commands

This chapter describes the Program Execution Analysis commands.

CHAPTER 5 Memory/Register Operation Commands

This chapter describes the Memory/Register Operation commands.

CHAPTER 6 Line Assemble and Disassemble Commands

This chapter describes the Line Assemble and Disassemble commands.

CHAPTER 7 Load and Save Commands

This chapter describes the Load and Save commands.

CHAPTER 8 Source File/Symbol Commands

This chapter describes the Source File/Symbol commands.

CHAPTER 9 Command Procedure Commands

This chapter describes the Command Procedure commands.

CHAPTER 10 Replacement Commands

This chapter describes the Replacement commands.

CHAPTER 11 Utility Commands

This chapter describes the Utility commands.

CHAPTER 12 Task Debug Commands

This chapter describes the Task Debug commands.

CHAPTER 13 Control Commands

This chapter describes the Control commands.

CHAPTER 14 Built-in Variables and Functions

This chapter describes the Built-in Variables and Functions.

APPENDIX

These appendixes describe the Manager-Related Messages, Error Message for Debuggers, and Execution Suspension Messages List.

- The contents of this document are subject to change without notice.
  Customers are advised to consult with sales representatives before ordering.
- The information, such as descriptions of function and application circuit examples, in this document are presented solely for the purpose of reference to show examples of operations and uses of FUJITSU SEMICONDUCTOR device; FUJITSU SEMICONDUCTOR does not warrant proper operation of the device with respect to use based on such information. When you develop equipment incorporating the device based on such information, you must assume any responsibility arising out of such use of the information. FUJITSU SEMICONDUCTOR assumes no liability for any damages whatsoever arising out of the use of the information.
- Any information in this document, including descriptions of function and schematic diagrams, shall not be construed as license of the use or exercise of any intellectual property right, such as patent right or copyright, or any other right of FUJITSU SEMICONDUCTOR or any third party or does FUJITSU SEMICONDUCTOR warrant non-infringement of any third-party's intellectual property right or other right by using such information. FUJITSU SEMICONDUCTOR assumes no liability for any infringement of the intellectual property rights or other rights of third parties which would result from the use of information contained herein.
- The products described in this document are designed, developed and manufactured as contemplated for general use, including without limitation, ordinary industrial use, general office use, personal use, and household use, but are not designed, developed and manufactured as contemplated (1) for use accompanying fatal risks or dangers that, unless extremely high safety is secured, could have a serious effect to the public, and could lead directly to death, personal injury, severe physical damage or other loss (i.e., nuclear reaction control in nuclear facility, aircraft flight control, air traffic control, mass transport control, medical life support system, missile launch control in weapon system), or (2) for use requiring extremely high reliability (i.e., submersible repeater and artificial satellite).
  Please note that FUJITSU SEMICONDUCTOR will not be liable against you and/or any third party for any claims or damages arising in connection with above-mentioned uses of the products.
- Any semiconductor devices have an inherent chance of failure. You must protect against injury, damage or loss from such failures by incorporating safety design measures into your facility and equipment such as redundancy, fire protection, and prevention of over-current levels and other abnormal operating conditions.
- Exportation/release of any products described in this document may require necessary procedures in accordance with the regulations of the Foreign Exchange and Foreign Trade Control Law of Japan and/or US export control laws.
- The company names and brand names herein are the trademarks or registered trademarks of their respective owners.

Copyright© 2004-2011 FUJITSU SEMICONDUCTOR LIMITED All rights reserved.

■ Command Reference Notation Format

The command reference notation format is given below.

Command name

■ Debugger

■ Format

■ Description

■ Example

Command name:

Name of command to be explained

Debuggers

Usable commands depend on the debugger type are explained. When using the emulator debugger, usable commands vary depending on the emulator.

[Debugger type]

Simulator : Simulator debugger

Emulator : Emulator debugger

Monitor : Monitor debugger

[Symbol for usable command]

◎: Can use command

○: Can use command except when instruction being executed

×: Cannot use command

—: There is no debugger

Format

The format, parameters, and command qualifiers of the command are explained. Enter the command in this format.

The following items are described in "● parameter".

- "(Default n-adic number)" shows the base number handled when prefix is omitted.
- For details of "(address formula)", refer to "2.1.5 Address Formula Specification" in "SOFTUNE Workbench Operation Manual".

Description

The command function is explained.

Example

Command coding example. This example may differ slightly from the actual coding

CONTENTS

CHAPTER 1 Environment Setup Commands 1

1.1 INITIALIZE 3
1.2 EXIT 4
1.3 RESET 5
1.4 SET MODE 6
1.5 SHOW MODE 8
1.6 SET TIMERSCALE 9
1.7 SHOW TIMERSCALE 10
1.8 SET RADIX 11
1.9 SHOW RADIX 12
1.10 SET SOURCE 13
1.11 SHOW SOURCE 14
1.12 SHOW SYSTEM 15
1.13 SET MAP (type 1) 16
1.14 SET MAP (type 2) 18
1.15 SHOW MAP 19
1.16 CANCEL MAP 20
1.17 ENABLE VERIFYMODE 21
1.18 DISABLE VERIFYMODE 22
1.19 SHOW VERIFYMODE 23
1.20 SET INPORT 24
1.21 SHOW INPORT 26
1.22 CANCEL INPORT 27
1.23 SET OUTPORT 28
1.24 SHOW OUTPORT 30
1.25 CANCEL OUTPORT 31
1.26 SET INTERRUPT 32
1.27 SHOW INTERRUPT 33
1.28 CANCEL INTERRUPT 34
1.29 SET VECTOR 35
1.30 SHOW VECTOR 36
1.31 ENABLE WATCHDOG 37
1.32 DISABLE WATCHDOG 38
1.33 SHOW WATCHDOG 39
1.34 SET WATCH 40
1.35 CANCEL WATCH 42

CHAPTER 2 Program Execution Commands 43

2.1 GO 44
2.2 SET GO 46
2.3 SHOW GO 47
2.4 ABORT 48
2.5 STEP 49

2.6 SET STEP 51
2.7 SHOW STEP 53
2.8 CALL 54
2.9 CLEAR CALL 56
2.10 SHOW STATUS 57

CHAPTER 3 Break/Event Control Command 59
3.1 SET BREAK (type 1) 61
3.2 SET BREAK (type 2) 64
3.3 SET BREAK (type 3) 65
3.4 SET BREAK (type 4) 66
3.5 SHOW BREAK 67
3.6 CANCEL BREAK 69
3.7 ENABLE BREAK 70
3.8 DISABLE BREAK 71
3.9 SET DATABREAK (type 1) 72
3.10 SET DATABREAK (type 2) 74
3.11 SHOW DATABREAK 76
3.12 CANCEL DATABREAK 77
3.13 ENABLE DATABREAK 78
3.14 DISABLE DATABREAK 79
3.15 SET EVENT 80
3.16 SHOW EVENT 82
3.17 CANCEL EVENT 83
3.18 ENABLE EVENT 84
3.19 DISABLE EVENT 85
3.20 SET SEQUENCE 86
3.21 SHOW SEQUENCE 88
3.22 CANCEL SEQUENCE 91
3.23 ENABLE SEQUENCE 93
3.24 DISABLE SEQUENCE 94
3.25 SET DELAY 95
3.26 SHOW DELAY 96

CHAPTER 4 Program Execution Analysis Commands 97
4.1 SET MULTITRACE 99
4.2 SHOW MULTITRACE 100
4.3 CLEAR MULTITRACE 102
4.4 ENABLE MULTITRACE 103
4.5 DISABLE MULTITRACE 104
4.6 SEARCH MULTITRACE 105
4.7 SET PERFORMANCE 107
4.8 SHOW PERFORMANCE 109
4.9 CLEAR PERFORMANCE 111
4.10 SET COVERAGE 112
4.11 SHOW COVERAGE 113
4.12 CANCEL COVERAGE 116

4.13 CLEAR COVERAGE 117
4.14 ENABLE COVERAGE 118
4.15 DISABLE COVERAGE 119
4.16 SET SAMPLING 120
4.17 SHOW SAMPLING 121
4.18 SHOW CALLS 122
4.19 SHOW TIMER 123
4.20 CLEAR TIMER 125
4.21 SET TRACE 126
4.22 SHOW TRACE (type 1) 127
4.23 SHOW TRACE (type 2) 129
4.24 SHOW TRACE (type 3) 130
4.25 CLEAR TRACE 132
4.26 ENABLE TRACE 133
4.27 DISABLE TRACE 134
4.28 SEARCH TRACE 135
4.29 SET RAMMONITOR 137
4.30 SHOW RAMMONITOR 139
4.31 CANCEL RAMMONITOR 140
4.32 ENABLE RAMMONITOR 142
4.33 DISABLE RAMMONITOR 143

CHAPTER 5 Memory/Register Operation Commands 145
5.1 EXAMINE 146
5.2 ENTER 148
5.3 SET MEMORY 150
5.4 SHOW MEMORY 152
5.5 SEARCH MEMORY 155
5.6 SET REGISTER 157
5.7 SHOW REGISTER 158
5.8 COMPARE 160
5.9 FILL 161
5.10 MOVE 162
5.11 DUMP 163
5.12 COPY 165
5.13 VERIFY 166

CHAPTER 6 Line Assemble and Disassemble Commands 167
6.1 ASSEMBLE 168
6.2 DISASSEMBLE 169

CHAPTER 7 Load and Save Commands 171
7.1 LOAD 172
7.2 SAVE 175

CHAPTER 8 Source File/Symbol Commands 177
8.1 LIST 178
8.2 SET PATH 180
8.3 SHOW PATH 181
8.4 SHOW SCOPE 182
8.5 UP 183
8.6 DOWN 184

CHAPTER 9 Command Procedure Commands 185
9.1 BATCH 186
9.2 QUIT 188

CHAPTER 10 Replacement Commands 189
10.1 SET ALIAS 190
10.2 SHOW ALIAS 191
10.3 CANCEL ALIAS 192
10.4 SET VARIABLE 193
10.5 SHOW VARIABLE 195
10.6 CANCEL VARIABLE 197

CHAPTER 11 Utility Commands 199
11.1 SET LOGGING 200
11.2 SHOW LOGGING 202
11.3 CANCEL LOGGING 203
11.4 ENABLE LOGGING 204
11.5 DISABLE LOGGING 205
11.6 PRINTF 206
11.7 SET OUTPUT 208
11.8 SHOW OUTPUT 209

CHAPTER 12 Task Debug Commands 211
12.1 SHOW OBJECT 212

CHAPTER 13 Control Commands 215
13.1 IF 216
13.2 REPEAT 218
13.3 WHILE 219
13.4 BREAK 220

CHAPTER 14 Built-in Variables and Functions 221
14.1 %CALL 222
14.2 %ERRNUM 223
14.3 %ENTRY 224
14.4 %STKTOP 225
14.5 %RADIX 226
14.6 %SCPADR 227

14.7 %LOADNUM 228
14.8 %BIT, %B, %W, %L, %S, %D 229
14.9 %STRGET 230
14.10 %STRSTR 231
14.11 %STRCMP 232
14.12 %STRLEN 233
14.13 %STRCAT 234
14.14 %SYMLEN 235
14.15 %TOVAL 236
14.16 %TOSTR 237
14.17 %EVAL 238

APPENDIX 239
APPENDIX A Manager-Related Messages 240
APPENDIX B Error Message for Debuggers 263
APPENDIX C Execution Suspension Messages List 298

INDEX..... 305

CHAPTER 1

Environment Setup Commands

This chapter describes the Environment Setup commands.

1.1 INITIALIZE

1.2 EXIT

1.3 RESET

1.4 SET MODE

1.5 SHOW MODE

1.6 SET TIMERSCALE

1.7 SHOW TIMERSCALE

1.8 SET RADIX

1.9 SHOW RADIX

1.10 SET SOURCE

1.11 SHOW SOURCE

1.12 SHOW SYSTEM

1.13 SET MAP (type 1)

1.14 SET MAP (type 2)

1.15 SHOW MAP

1.16 CANCEL MAP

1.17 ENABLE VERIFYMODE

1.18 DISABLE VERIFYMODE

1.19 SHOW VERIFYMODE

1.20 SET INPORT

1.21 SHOW INPORT

1.22 CANCEL INPORT
1.23 SET OUTPORT
1.24 SHOW OUTPORT
1.25 CANCEL OUTPORT
1.26 SET INTERRUPT
1.27 SHOW INTERRUPT
1.28 CANCEL INTERRUPT
1.29 SET VECTOR
1.30 SHOW VECTOR
1.31 ENABLE WATCHDOG
1.32 DISABLE WATCHDOG
1.33 SHOW WATCHDOG
1.34 SET WATCH
1.35 CANCEL WATCH

1.1 INITIALIZE

The INITIALIZE command initializes the debugger.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

INITIALIZE

■ Description

The INITIALIZE command initializes the debugger.

This initialization nullifies all settings other than macro, alias and debug variable.

■ Example

```
>INITIALIZE
```

1.2 EXIT

The EXIT command terminates the debugger.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

EXIT

■ Description

The EXIT command terminates the debugger.

■ Example

```
>EXIT
```

1.3 RESET

The RESET command inputs the reset signal to the MCU.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

RESET

■ Description

The RESET command inputs the reset signal to the MCU.

■ Example

```
>RESET
```

1.4 SET MODE

The SET MODE command sets the event mode as follows.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET MODE

● Command qualifiers

Set event mode.

/NORMAL (default at start-up)

Sets event mode to NORMAL mode.

/MULTITRACE

Sets event mode to MULTITRACE mode.

/PERFORMANCE

Sets event mode to PERFORMANCE mode.

■ Description

The SET MODE command sets the event mode:

- NORMAL mode

  The event function is used for control by a sequencer. Command setting related to SEQUENCE, DELAY, and TRACE is enabled.

- MULTITRACE mode

  The event function is used for multi-tracing. Command setting related to MULTITRACE is enabled.

- PERFORMANCE mode

  The event function is used for measuring performance. Command setting related to PERFORMANCE is enabled.

The commands related to EVENT can be used in all modes, each mode has different values. If a mode is changed, the value will return to the value previously set in the mode.

A mode change will also clear the single-trace, multitrace, and performance buffers.

The start-up is "/NORMAL".

■ Example

```
>SET MODE /MULTITRACE
```

1.5 SHOW MODE

The SHOW MODE command displays the setting state of the event mode.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW MODE

■ Description

The SHOW MODE command displays the setting state of the event mode.

■ Example

```
>SHOW MODE
event mode: normal
```

1.6 SET TIMERSCALE

The SET TIMERSCALE command sets the minimum measurement unit of a timer.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET TIMERSCALE

● Command qualifiers

Sets the minimum measurement unit of a timer.

/1 μs (default when omitted)

Sets minimum measurement unit of timer to 1 μs.

/100 ns

Sets minimum measurement unit of timer to 100 ns.

■ Description

The SET TIMERSCALE command sets the minimum measurement unit of a timer.

The unit can be set to either 1 μs or 100 ns. The start-up is 1 μs. This setting will determine the minimum measurement units of the following timers:

- Sequencer timer
- Emulation timer
- Performance measurement timer

■ Example

```
>SET TIMERSCALE/100ns
```

1.7 SHOW TIMERSCALE

The SHOW TIMERSCALE command displays the minimum measurement unit of a timer.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW TIMERSCALE

■ Description

The SHOW TIMERSCALE command displays the minimum measurement unit of a timer.

■ Example

```
>SHOW TIMERSCALE
Timer scale: 100 ns
```

1.8 SET RADIX

The SET RADIX command sets default base number.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SET RADIX

● Command qualifiers

/BINARY

Sets default base number to binary number.

/OCTAL

Sets default base number to octal number.

/DECIMAL

Sets default base number to decimal number.

/HEXADECIMAL (default)

Sets default base number to hexadecimal number.

■ Description

The SET RADIX command sets default base number.

■ Example

```
>SET RADIX/HEXADECIMAL
```

1.9 SHOW RADIX

The SHOW RADIX command displays the current base number.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW RADIX

■ Description

The SHOW RADIX command displays the current base number.

■ Example

```
>SHOW RADIX
default radix: hexadecimal
```

1.10 SET SOURCE

The SET SOURCE command sets source line display mode.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SET SOURCE

● Command qualifiers

Source line display mode

/DISPLAY (default at start-up)

Sets mode in which source lines displayed.

/NODISPLAY

Sets mode in which source lines not displayed.

■ Description

When the disassemble list is displayed, the SET SOURCE command sets whether or not to display the added source line.

When the debugger is started, the mode in which source lines are displayed is set.

■ Example

```
>SET SOURCE/DISPLAY
```

1.11 SHOW SOURCE

The SHOW SOURCE command displays the source line display mode set by the SET SOURCE command.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW SOURCE

■ Description

The SHOW SOURCE command displays the source line display mode set by the SET SOURCE command.

■ Example

```
>SHOW SOURCE
source mode: display
```

1.12 SHOW SYSTEM

The SHOW SYSTEM command displays system information.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW SYSTEM

■ Description

The SHOW SOURCE command displays system information.

What is displayed varies depending on the debugger type.

For details, refer to the following sections of "SOFTUNE WORKBENCH USER'S MANUAL".

Simulator debugger	: "2.1.11 Checking State of Debugger"
Emulator debugger (MB2141)	: "2.2.12 Checking State of Debugger"
Emulator debugger (MB2146-09/09A/09B)	: "2.3.8 Checking State of Debugger"
Emulator debugger (MB2146-08)	: "2.4.8 Checking State of Debugger"
Emulator debugger (MB2146-07)	: "2.5.9 Checking State of Debugger"
Monitor debugger	: "2.5.5 Checking State of Debugger"

■ Example

[Emulator debugger (MB2141)]

```
F2MC-8L/8FX Family SOFTUNE Workbench V30L33
  Debugger type          = Emulator Debugger
  MCU type               = MB89W625
  VCpu dll name          = C:\Softune\bin\wv896e1.dll
  VCpu dll version       = V01L04
  Monitor version        = V03L05
  MCU frequency          = 10.000 MHz
  Communication device   = LAN
  Host name              = 127.0.0.1
```

1.13 SET MAP (type 1)

The SET MAP command sets a memory space area type and access attribute.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET MAP {address | address-range}

● Parameters

address (address formula)

Specify the memory address where access attribute to be set.

address-range (address formula)

Specify the memory area where access attribute to be set.

● Command qualifiers

- Specifying access attribute

/READ

Enables data read access.

The code access is also handled as READ.

/WRITE

Enables data write access.

/CODE

Enables code read access.

If command qualifier is omitted, /READ/WRITE is set.

- Specifying area type

/USER [MB2141]

Sets area to user memory area.

/EMULATION (default when omitted) [MB2141]

Sets area to emulation memory area.

■ Description

The SET MAP command sets a memory space area type and access attribute.

- Simulator Debugger

  Up to 31 areas can be set. (Enabled to specify in byte)

  When the load module file is loaded by the LOAD command, appropriate access attributes are automatically set according to the file information.

- Emulator Debugger (MB2141)
  - User memory area (/USER)

    Up to 20 areas can be specified with the emulation area in one byte.

    There is no restriction on the size of one area.
  - Emulation area (/EMULATION)

    Up to 20 areas can be specified with the user memory area in one byte.

    There is no restriction on the size of one area.

■ Example

```
>SET MAP/READ/WRITE 1000..1FFF
```

1.14 SET MAP (type 2)

The SET MAP command sets an undefined area access attributes.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET MAP {/GUARD | /NOGUARD}

● Command qualifiers

Setting undefined area access attributes

/GUARD (default when internal ROM provided)

Disables access to undefined area.

/NOGUARD (default when internal ROM not provided)

Enables full access to undefined area.

■ Description

The SET MAP command sets an access attributes to undefined area.

Either the command qualifier for allowing any access (/NOGUARD) or for disallowing access (/GUARD) can be specified for an undefined area.

■ Example

```
>SET MAP/GUARD
```

1.15 SHOW MAP

The SHOW MAP command displays the set memory space access attributes.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW MAP

■ Description

The SHOW MAP command displays the set memory space access attributes.

■ Example

```
>SHOW MAP
  address         attribute              type
  0000 .. 027F    code   read   write    user
  C000 .. FFFF    code   read            emulation
 ------------------------------------------------------------------
   undefined area : guard
   setup possibility : user = 19    emulation = 19
```

1.16 CANCEL MAP

The CANCEL MAP command assigns the undefined attribute to the specified address area.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CANCEL MAP {address | address-range}

● Parameters

address (address formula)

Specify the address where undefined attribute to be assigned.

address-range (address formula)

Specify the address range where undefined attribute to be assigned.

● Command qualifier

/ALL

Assigns undefined attribute to all set maps.

■ Description

The CANCEL MAP command assigns the undefined attribute to the specified address area.

■ Example

```
>CANCEL MAP/ALL
```

1.17 ENABLE VERIFYMODE

The ENABLE VERIFYMODE command enables the verify operation used when memory is written by a command.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

ENABLE VERIFYMODE

■ Description

The ENABLE VERIFYMODE command enables the verify operation used when memory is written by a command.

The verify operation is enabled when the debugger is started.

■ Example

```
>ENABLE VERIFYMODE
```

1.18 DISABLE VERIFYMODE

The DISABLE VERIFYMODE command disables the verify operation used when memory is written by a command.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

DISABLE VERIFYMODE

■ Description

The DISABLE VERIFYMODE command disables the verify operation used when memory is written by a command.

The verify operation is enabled when the debugger is started.

■ Example

```
>DISABLE VERIFYMODE
```

1.19 SHOW VERIFYMODE

The SHOW VERIFYMODE command displays the status of the verify mode.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW VERIFYMODE

■ Description

The SHOW VERIFYMODE command displays the status of the verify mode (mode in which verify operation enabled or disabled is displayed when memory is written by a command).

■ Example

```
>SHOW VERIFYMODE
verify mode enable
```

1.20 SET INPORT

The SET INPORT command specifies data input to a specified port.

■ Debugger

Simulator		○
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET INPORT port-address, mask-data, data-input-source [, cycle-count]

● Parameters

port-address

Specify the port address.

mask-data

Specify the mask data.

Only 1 mask data bit can be used as port data.

data-input-source

Specify the data input source (file name, input terminal) when program reads data from input port or when count of program instruction execution cycles exceeds specified cycle count.

Specify $TERMINAL as input terminal.

$TERMINAL cannot be specified as file name.

cycle-count (default decimal number)

Specify the count of program instruction execution cycles (H'1 to H'FFFFFFFF).

● Command qualifiers

Specifying access size

/BYTE (default when omitted)

Specifies port access when specified address accessed 8 bits.

/WORD

Specifies port access when specified address accessed 16 bits.

/LONG

Specifies port access when specified address accessed 32 bits.

/ASCII

Uses character codes of input data as input values.

When /ASCII is specified, the access size is always /BYTE.

■ Description

The SET INPORT command specifies data input to a port each time the program reads data from the specified port or each time the count of program instruction execution cycles exceeds the specified cycle count.

When cycle-count is not specified in the parameters, data is read from the specified data input source each time the program reads data from the input port.

When cycle-count is specified in the parameters, port contents are updated per fixed cycle irrespective of port access.

If data-input-source is a file, data input processing returns to the beginning of the file when the last data is entered.

If data-input-source is a input terminal ($TERMINAL), the dialog box for data input request is displayed when the set port is read-accessed. When this dialog box appears, enter the input data.

Up to 4096 port addresses can be simulated.

■ Example

```
>SET INPORT 0, 1F, INBUF0. DAT
>SHOW INPORT
address     bit pattern     size     cycle     input
0000        001F            byte     ------    INBUF0. DAT
004F        000F            ascii    ------    $terminal
```

1.21 SHOW INPORT

The SHOW INPORT command displays the data set by the SET INPORT command.

■ Debugger

Simulator		◎
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW INPORT

■ Description

The SHOW INPORT command displays the data set by the SET INPORT command.

■ Example

```
>SHOW INPORT
address     bit pattern     size     cycle     input
00FF        00FF            byte     ------    $terminal
004F        000F            ascii    ------    $terminal
```

1.22 CANCEL INPORT

The CANCEL INPORT command cancels simulation of specified port address.

■ Debugger

Simulator		○
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CANCEL INPORT [port-address [, ...]]

● Parameter

port-address

Specify the port address.

● Command qualifier

/ALL

Cancels all data set by SET INPORT command.

■ Description

The CANCEL INPORT command cancels simulation of specified port address.

■ Example

```
>CANCEL INPORT/ALL
```

1.23 SET OUTPORT

The SET OUTPORT command specifies data output to the specified port.

■ Debugger

Simulator		○
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET OUTPORT port-address, mask-data, data-output-destination

● Parameters

port-address

Specify the port address.

mask-data

Specify the mask data.

Only 1 mask data bit can be used as port data.

data-output-destination

Specify the data output destination (file name, output terminal) where data written to output port by program to be stored.

Specify $TERMINAL as the output terminal.

$TERMINAL cannot be specified as the file name.

The same file name cannot be used.

● Command qualifiers

Specifying access size

/BYTE (default when omitted)

Specifies port access when specified address accessed 8 bits.

/WORD

Specifies port access when specified address accessed 16 bits.

/LONG

Specifies port access when specified address accessed 32 bits.

/ASCII

When data-output-destination is $TERMINAL, the debugger converts the data output to the port

(regarded as ASCII codes) to characters and displays them on the screen.

When data-output-destination is a file, the debugger outputs binary codes as they are.

When /ASCII is specified, the access size is always /BYTE.

■ Description

The SET OUTPORT command specifies that data is to be stored in the specified data output destination each time the program writes data to the specified port.

If the disk becomes full when data-output-destination is a file, the debugger displays the error message and does not store the subsequent data in the file.

When data-output-destination is a output terminal ($TERMINAL), the debugger displays the data output to the port on the screen in hexadecimal notation.

However, when command qualifier /ASCII is specified, the debugger displays the output data in characters, because the debugger treats the output data as character codes.

Up to 4096 port addresses can be simulated.

■ Example

```
>SET OUTPORT 0, 3F, OU1. DAT
>SHOW OUTPORT
address       bit pattern       size       output
0000          003F              byte       OU1. DAT
0040          00FF              ascii      $TERMINAL
```

1.24 SHOW OUTPORT

The SHOW OUTPORT command displays the data set by the SET OUTPORT command.

■ Debugger

Simulator		◎
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW OUTPORT

■ Description

The SHOW OUTPORT command displays the data set by the SET OUTPORT command.

■ Example

```
>SHOW OUTPORT
address       bit pattern       size       output
0000          003F              byte       OU1. DAT
0040          00FF              ascii      $TERMINAL
```

1.25 CANCEL OUTPORT

The CANCEL OUTPORT command cancels the simulation of the specified port address.

■ Debugger

Simulator		○
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CANCEL OUTPORT [port-address [, ...]]

● Parameter

port-address

Specify the port address.

● Command qualifier

/ALL

Cancels simulation of port set by SET OUTPORT command.

■ Description

The CANCEL OUTPORT command cancels the simulation of the specified port address.

■ Example

```
>CANCEL OUTPORT/ALL
```

1.26 SET INTERRUPT

The SET INTERRUPT command sets interrupt-generation condition.

■ Debugger

Simulator		○
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET INTERRUPT interrupt-number, cycle-count

● Parameters

interrupt-number (default decimal number)

Specify the interrupt vector number.

For information about how many interrupts can be used, refer to the hardware manual.

cycle-count (default decimal number)

Specify the count of program instruction execution cycles (D'1 to D'4294967295).

● Command qualifier

/INTERVAL

Specifies cyclic generation of interrupts.

When omitted, single interrupt generation will be specified.

■ Description

When the program is executed for the specified count of instruction execution cycles or more, the SET INTERRUPT command causes the specified interrupt and cancels the interrupt-generation condition.

When /INTERVAL is specified, the specified interrupt is generated per specified count of instruction execution cycles during program execution.

The interrupt-generation condition is valid until it is cancelled by the CANCEL INTERRUPT command.

■ Example

```
>SET INTERRUPT 4, 12367
```

1.27 SHOW INTERRUPT

The SHOW INTERRUPT command displays the interrupt-generation condition set by the SET INTERRUPT command.

■ Debugger

Simulator		◎
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW INTERRUPT

■ Description

The SHOW INTERRUPT command displays the interrupt vector number, cycle count (decimal number), and interrupt type set by the SET INTERRUPT command in this order.

Interrupt type specifies any of the following:

shot: specifies single interrupt generation.

Interval: specifies repetitive interrupt generation.

■ Example

```
>SHOW INTERRUPT
no    cycle   kind
4     1258    shot
7     9823    interval
```

1.28 CANCEL INTERRUPT

The CANCEL INTERRUPT command cancels all the interrupt-generation conditions.

■ Debugger

Simulator		○
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CANCEL INTERRUPT [interrupt-number [, ...]]

● Parameter

interrupt-number (default decimal number)

Specify the interrupt vector number.

● Command qualifier

/ALL

Cancels all interrupt-generation conditions set by SET INTERRUPT command.

■ Description

The CANCEL INTERRUPT command cancels all the interrupt-generation conditions set by the SET INTERRUPT command.

■ Example

```
>CANCEL INTERRUPT/ALL
```

1.29 SET VECTOR

The SET VECTOR command sets the vector number data.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	×
Monitor		×

■ Format

SET VECTOR vector-number, address-value

● Parameters

vector-number

Specify the number of vector to be set.

address-value

Specify the starting address of routine corresponding to specified vector number.

■ Description

The SET VECTOR command sets the address value of the vector number in the specified area.

■ Example

```
>SET VECTOR 7,FF10
>SHOW VECTOR 7..7
Vector No.     Address     Symbol     Factor
7              FF10                   time-base timer
```

1.30 SHOW VECTOR

The SHOW VECTOR command displays vector number data.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SHOW VECTOR [vector-number-range]

● Parameter

vector-number-range

Specify the range of vector numbers to be displayed.

Specify range in "[starting-number..ending-number]" format.

■ Description

The SHOW VECTOR command displays vector number data.

If vector-number-range specifying is omitted, vector number display is started from the next vector number.

■ Example

```
>SHOW VECTOR 6..10
 Vector No.     address     Symbol     Factor
 6              0000                   16-bit timer/counter
 7              0000                   8-bit serial I/O #1
 8              0000                   8-bit serial I/O #2
 9              0000                   System reserved
10              ff10                   timebase timer
```

1.31 ENABLE WATCHDOG

The ENABLE WATCHDOG command enables a watchdog timer.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		×

■ Format

ENABLE WATCHDOG

■ Description

The ENABLE WATCHDOG command enables a watchdog timer.

■ Example

```
>ENABLE WATCHDOG
```

1.32 DISABLE WATCHDOG

The DISABLE WATCHDOG command disables a watchdog timer.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		×

■ Format

DISABLE WATCHDOG

■ Description

The DISABLE WATCHDOG command disables a watchdog timer.

■ Example

```
>DISABLE WATCHDOG
```

1.33 SHOW WATCHDOG

The SHOW WATCHDOG command displays the enabled/disabled state of a watchdog timer.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		×

■ Format

SHOW WATCHDOG

■ Description

The SHOW WATCHDOG command displays the enabled/disabled state of a watchdog timer.

■ Example

```
>SHOW WATCHDOG
watchdog: enable
```

1.34 SET WATCH

The specified variable is registered to the watch window.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		○

■ Format

SET WATCH variable [, watch-window-number]

● Parameters

variable

Specifies variables or expressions to be set in the watch window.

watch-window-number

Specify the number (1 to 4) of the watch window to which variables are added. When omitted, Watch Window 1 will be specified.

● Command qualifiers

Base Number

/BINARY

Specify that variable values will be displayed in binary.

/ OCTAL

Specify that variable values will be displayed in octal.

/ DECIMAL

Specify that variable values will be displayed in decimal.

/ HEXADECIMAL

Specify that variable values will be displayed in hexadecimal.

Watch Mode

/ AUTOMATIC

Interprets variables in the watch mode set in the debug environment.

If the setting in the debug environment is changed, the watch mode for variables is changed as well.

/ C

Interprets variables as C language expressions.

/ASSEMBLER

Interprets variables as assembler expressions.

Data Size

/BYTE

Specify that display will be provided with 8-bit length in assembler mode.

/WORD

Specify that display will be provided with 16-bit length in assembler mode.

/LONG

Specify that display will be provided with 32-bit length in assembler mode.

/SINGLE

Specify that display will be provided with single-precision floating-point number in assembler mode.

/DOUBLE

Specify that display will be provided with double-precision floating-point number in assembler mode.

Individual monitoring setting

/MONITORING (Default when omitted)

Sets the individual monitoring settings for variables to ON.

Can specify this command qualifier in the simulator debugger and the emulator debugger (MB2141).

NOMONITORING

Sets the individual monitoring settings for variables to OFF.

Can specify this command qualifier in the simulator debugger and the emulator debugger (MB2141).

■ Description

Sets variables in the specified watch window. If variables already set are specified, two or more variables of the same name are set.

If the command qualifier for a base number, watch mode, or data size is omitted, the setting specified in the debug environment is effective.

The specified data size will be valid only when the setting of /ASSEMBLER is specified.

For a setting of a watch point, previous information is restored when the Debugger is started. If a watch point is set in the batch file when the Debugger is started, once delete all the watch points by CANCEL WATCH/ALL.

■ Example

```
>SET WATCH strsym.a,1
>SET WATCH/HEXADECIMAL/ASSEMBLER/WORD/NOMONITORING LABEL1,1
```

1.35 CANCEL WATCH

The CANCEL WATCH command deletes specified variables from the watch window.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

CANCEL WATCH variable [, watch-window-number]

CANCEL WATCH /ALL [, watch-window-number]

● Parameters

variables

Specifies variables or expressions to be deleted from the watch window.

watch-window-number

Specifies the number (1 to 4) of the watch window from which variables are deleted.

When omitted, Watch Window 1 is specified.

● Command qualifier

/ ALL

Deletes all watch points from the specified window.

■ Description

Deletes variables from the specified watch window.

If two or more variables of the same name exist, only the variable that first appears will be deleted.

■ Example

```
>CANCEL WATCH flag,1
>CANCEL WATCH/ALL 2
```

CHAPTER 2

Program Execution Commands

This chapter describes the Program Execution commands.

2.1 GO
2.2 SET GO
2.3 SHOW GO
2.4 ABORT
2.5 STEP
2.6 SET STEP
2.7 SHOW STEP
2.8 CALL
2.9 CLEAR CALL
2.10 SHOW STATUS

2.1 GO

The GO command executes the program from the specified starting address.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

GO [starting-address] [, break-address1] [, break-address2]

● Parameters

starting-address (address formula)

Specify the address at which program execution started.

break-address (address formula)

Specify the address at which program execution stopped.

This parameter is invalid if /RETURN or /NOWAIT is specified.

For only the following debugger type, "break-address2" can be specified.

- Simulator debugger
- Emulator debugger (MB2141)

●Command qualifiers

Return setting

/RETURN

Executes program from function currently being executed to parent function return location.

Only programs coded in C language can use this function.

The optimized program may not be stopped normally.

Setting interrupt mask

/MASK

Masks interrupt.

When the debugger type is only the simulator debugger or the emulator debugger (MB2141), this command qualifier can be specified.

/NOMASK

Does not mask interrupt.

When the debugger type is only the simulator debugger or the emulator debugger (MB2141), this command qualifier can be specified.

Selection of command input during execution

/WAIT (default when not specified)

Once the user program is run, other commands are not accepted until the program stops.

/NOWAIT

Commands can be input after the user program is run without waiting for the program to stop.

This qualifier cannot be specified if /RETURN is specified.

[MB2141]

Trace control

/ENABLETRACE

Enables trace function at start of program execution.

/DISABLETRACE

Disables trace function at start of program execution.

■ Description

The GO command executes the program from the specified starting address.

If starting-address specifying is omitted, the program is executed from the address indicated by the current program counter.

The break address set by the GO command is automatically deleted when program execution is stopped.

The command qualifiers, /ENABLETRACE and /DISABLETRACE, are specified for trace control by a sequencer.

If a command qualifiers is omitted, program execution will start as set by the SET GO command.

■ Example

```
>GO power$20
Break at main$10
>GO power$20, main$5
```

2.2 SET GO

The SET GO command specifies the execution conditions for the GO command as command qualifiers are omitted.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET GO

● Command qualifiers

Setting interrupt mask

/MASK

Masks interrupt.

/NOMASK (default at start-up)

Does not mask interrupt.

[MB2141]

Trace control

/ENABLETRACE (default at start-up)

Enables trace function at start of program execution.

/DISABLETRACE

Disables trace function at start of program execution.

■ Description

The SET GO command specifies the execution conditions for the GO command as command qualifiers are omitted.

■ Example

```
>SET GO /MASK
>GO
```

2.3 SHOW GO

The SHOW GO command displays the current execution conditions (SET GO command settings) for the GO command.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW GO

■ Description

The SHOW GO command displays the current execution conditions (SET GO command settings) for the GO command.

■ Example

```
>SHOW GO
Interrupt mask:  nomask
Trace control:   enable
```

2.4 ABORT

The ABORT command stops the currently executing program.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

ABORT

● Parameters

None

● Command qualifiers

None

■ Description

The ABORT command stops the currently executing program.

This command is valid in the following situations.

- If /NOWAIT is specified in the GO command
- If the continuous execution button is pressed in the execution toolbar

■ Example

```
> ABORT
```

Note:

If this command is input while the user program is stopped, the message "This command cannot be used while the MCU is stopped" is displayed.

2.5 STEP

The STEP command executes the program in units of source lines or machine instructions according to the condition set by the STEP command.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

STEP [step-count]

● Parameter

step-count (default decimal number)

Specify the count of STEP command executed (H'1 to H'FFFFFFFF).

If step-count specifying is omitted, the count of times is 1.

● Command qualifiers

- Step unit specification

/INSTRUCTION

Executes program in the unit of a machine instruction.

/LINE

Executes program in units of source lines.

/AUTOMATIC (default at start-up)

Automatically changes execution unit according to source window display mode as follows:

- When the source window display mode is the source line display mode, the program is executed in units of source lines (/LINE).
- When the source window display mode is another display mode, the program is executed in units of machine instructions (/INSTRUCTION).

- Step operation specification

/INTO

Executes program for each step in called function, subroutine, or interrupt handler.

/OVER

Executes the following instructions as one step:

- Function call when /LINE is specified
- Subroutine call instruction (e.g. CALL) when /INSTRUCTION is specified
- Software interrupt instruction (e.g. INT) when /INSTRUCTION is specified

- Specification of the interrupt mask

/MASK

Masks interrupt.

When the debugger type is only the simulator debugger or the emulator debugger (MB2141), this command qualifier can be specified.

/NOMASK (default at start-up)

Does not mask interrupt.

When the debugger type is only the simulator debugger or the emulator debugger (MB2141), this command qualifier can be specified.

■ Description

The STEP command executes the program in units of source lines or machine instructions according to the condition set by the SET STEP command.

When the Softune workbench is started, the step Invalid condition is AUTOMATIC, INTO.

■ Example

```
>STEP
>STEP/INSTRUCTION
```

2.6 SET STEP

The SET STEP command specifies the step execution condition when no command qualifier is specified in the STEP command.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SET STEP

● Command qualifiers

Specification of the STEP unit

/INSTRUCTION

Executes program in the unit of a machine instruction.

/LINE

Executes program in units of source lines.

/AUTOMATIC (default at start-up)

Automatically changes execution unit according to source window display mode as follows:

- When the source window display mode is the source line display mode, the program is executed in units of source lines (/LINE).
- When the source window display mode is another display mode, the program is executed in units of machine instructions (/INSTRUCTION).

Specification of the STEP operation

/INTO (default at start-up)

Executes program for each step in called function, subroutine, or interrupt handler.

/OVER

Executes the following instructions as one step:

- Function call when /LINE is specified
- Subroutine call instruction (e.g. CALL) when /INSTRUCTION is specified
- Software interrupt instruction (e.g. INT) when /INSTRUCTION is specified

Specification of the interrupt mask

/MASK

Masks interrupt

When the debugger type is only the simulator debugger or the emulator debugger (MB2141), this command qualifier can be specified.

/NOMASK (default at start-up)

Does not mask interrupt.

When the debugger type is only the simulator debugger or the emulator debugger (MB2141), this command qualifier can be specified.

■ Description

The SET STEP command specifies the step execution condition when no command qualifier is specified in the STEP command.

SET STEP/AUTOMATIC/INTO/NOMASK is specified when SOFTUNE Workbench is executed.

■ Example

```
>SET STEP/INSTRUCTION
```

2.7 SHOW STEP

The SHOW STEP command displays the step execution condition of the current STEP command.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW STEP

■ Description

The SHOW STEP command displays the step execution condition of the current STEP command.

■ Example

```
>SHOW STEP
step mode: instruction, into
```

2.8 CALL

The CALL command executes the specified function and displays a return value.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		×

■ Format

CALL function-name ([argument [, ...]])

● Parameters

function-name

Specify the name of function to be called.

argument

Compiles with the arguments in C language.

However, structures, unions and class cannot be specified as variable names.

● Command qualifiers

/DISPLAY (Default at start-up)

Sets a return value displayed.

/NODISPLAY

Sets a return value not displayed.

■ Description

The CALL command executes the specified function and displays a return value. (However, if the return value is structure, union, or class type, an error occurs).

The CALL command can be used only when the program coded in C language is compiled with debug information.

If a break point is reached when a function is being executed by the CALL command, the program breaks at that position.

CALL command execution is continued by subsequently restarting program execution with the GO command.

To suspend CALL command execution, use the CLEAR CALL command.

The CALL command cannot be nested.

The register and flag values before the function is called are retained. These values are restored to the original values after the function has been executed.

The argument of the specified function is evaluated and executed in dummy argument type.

If the count of specified actual arguments is greater than that of dummy arguments, extra actual arguments are evaluated in int type.

The return value is set in built-in variable %CALL.

The CALL command sets a break point at the address indicated by the current program counter and sets the return address at the address so that control will return to the break point. The command then calls the function.

For this reason, if the function executed by the CALL command accidentally passes the address indicated by the current program counter, the program will break in the middle of the function.

In this case, the following message is displayed:

Break at address by Invalid call termination

CALL command execution is continued by restarting program execution with the GO command.

■ Example

```
>CALL debug (cmd, p)
return value is H'0001
```

2.9 CLEAR CALL

The CLEAR CALL command cancels the CALL command and restores the status set before the register is called.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		×

■ Format

CLEAR CALL

■ Description

The CLEAR CALL command cancels the CALL command and restores the status set before the register is called.

■ Example

```
>CALL debug (cmd, p)
Break at FF20 by break point
>CLEAR CALL
```

2.10 SHOW STATUS

The SHOW STATUS command displays the MCU execution status.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW STATUS

■ Description

- During the program running

  The SHOW STATUS command displays the MCU execution status.
- During the program stopping

  The SHOW STATUS command displays the break factor of the immediately-preceding program execution.

The MCU execution status is displayed as follows.

Display contents		Debugger		
			Emulator	
Display result	MCU execution status	Simulator	MB2141 MB2146-09 MB2146-08 MB2146-07	Monitor
Break	Break (user state)	○	○	○
Execute	Execute (debug state)	○	○	○
Stop	Stop mode	○	×	×
Sleep	Sleep mode	○	×	×

○ : Displayed, × : Not displayed

For the break factor, refer to "APPENDIX C Execution Suspension Messages List".

■ Example

- MCU breaking

```
>SHOW STATUS
MCU status : Break at startup.asm$64(H'C12F) by breakpoint
```

[MB2146-07]

```
>SHOW STATUS
MCU status : Break at startup.asm$62(H'B00B) by hardware breakpoint
```

- MCU executing

```
>SHOW STATUS
MCU status : Execute
```

CHAPTER 3

Break/Event Control Command

This chapter describes the Break/Event Control commands.

3.1 SET BREAK (type 1)
3.2 SET BREAK (type 2)
3.3 SET BREAK (type 3)
3.4 SET BREAK (type 4)
3.5 SHOW BREAK
3.6 CANCEL BREAK
3.7 ENABLE BREAK
3.8 DISABLE BREAK
3.9 SET DATABREAK (type 1)
3.10 SET DATABREAK (type 2)
3.11 SHOW DATABREAK
3.12 CANCEL DATABREAK
3.13 ENABLE DATABREAK
3.14 DISABLE DATABREAK
3.15 SET EVENT
3.16 SHOW EVENT
3.17 CANCEL EVENT
3.18 ENABLE EVENT
3.19 DISABLE EVENT
3.20 SET SEQUENCE
3.21 SHOW SEQUENCE

3.22 CANCEL SEQUENCE
3.23 ENABLE SEQUENCE
3.24 DISABLE SEQUENCE
3.25 SET DELAY
3.26 SHOW DELAY

3.1 SET BREAK (type 1)

The SET BREAK command sets a break point at the specified break address.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SET BREAK break-address [, pass-count] [, {command : command ... }]

● Parameters

break-address (address formula)

Specify the address at which break point set.

pass-count (default decimal number)

Specify the number of times break point to be hit (1 to 65535).

Program execution specifying is stopped when this number of times is reached.

If pass-count is omitted, 1 is assumed.

This function is valid when the debugger type is only a simulator debugger.

command

Specify the command list for executing when the break address is hit. Two or more commands can be specified by using the colon.

This function is valid when the debugger type is only a simulator debugger.

● Command qualifiers (Only the simulator debugger)

- Program execution specification after processing command list

/ BREAK (default when omitted)

After the command list is processed, the instruction execution is stopped when the break point is hit.

This function is valid when the debugger type is only a simulator debugger.

/ NOBREAK

After the command list is processed, the instruction execution is restarted when the break point is hit.

This function is valid when the debugger type is only a simulator debugger.

- Specification of break point type

/ HARD (default at start-up)

The break point by hardware is set. This function is valid when the debugger type is only a emulator debugger (MB2146-08/MB2146-07).

/ SOFT

The break point by software is set. This function is valid when the debugger type is only a emulator debugger (MB2146-08/MB2146-07).

■ Description

The SET BREAK command sets a break point at the specified break address.

The maximum settable count of break points to be specified is as follows:

Debugger		Break point	
		Hardware	Software
Emulator	MB2141	65535 (*1) (*2)	
	MB2146-09/09A/09B	256 (*1) (*3)	
	MB2146-08	3	256 (*3)
	MB2146-07	3	256 (*3)
Simulator		65535 (*1)	
Monitor		2	

*1: There is no distinction between hardware and software.

*2: 65535 breakpoints or less can be set in the area where the debugging area is set as a code break area. This area is set by the SET DEBUG command.

*3: When the breakpoint is set, the code is rewritten.

■ Example

[Common for each debugger]

```
>SET BREAK FF02
```

[Simulator debugger]

```
>SET BREAK FF03, 3
>SET BREAK/NOBREAK main, 1, {SHOW   TRACE : SHOW TIMER}
```

[Emulator debugger (MB2146-08/MB2146-07)]

```
>SET BREAK/HARD FF02
```

Notes:

- The following execution type command cannot be specified for the command list.
  - GO
  - STEP
  - CALL
- The STUB function (restart command execution and instruction execution) is not executed in STEP/INTO and STEP/OVER. After the break point is hit, execution is stopped.
- A hardware break setting does not stop the program if one of the following operations is performed while a user program is executing with the emulator debugger (MB2146-08/MB2146-07).
  - Input the reset signal from the target board.
  - Select [Debug] - [Reset MCU] menu
- Specify the starting address of an instruction when a breakpoint is set by using the following debuggers. If the other address is specified, the program may run away.
  - Simulator debugger
  - Emulator debugger (MB2146-09/09A/09B)
  - Emulator debugger (MB2146-08)
  - Emulator debugger (MB2146-07)

3.2 SET BREAK (type 2)

The SET BREAK command sets a break point at the specified break address.

■ Debugger

Simulator		×
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET BREAK / DATAWATCH code-address

● Parameter

code-address (address formula)

Specify the address of code that serves as a data monitoring condition.

■ Description

The SET BREAK command sets the code break point for the data watch break function.

The data watch break function is used to stop the program when the program execution hits the code break point with the data break points hit state.

It is necessary to set the data break before using this command.

However, when this function is used, the data break function does not work.

■ Example

```
> SET BREAK/DATAWATCH  ff00
```

3.3 SET BREAK (type 3)

The SET BREAK command sets a break point at the specified break address.

■ Debugger

Simulator		×
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET BREAK / SEQUENCE code-address1, code-address2

● Parameters

code-address1 (address formula)

Specify the address 1 of code that serves as a sequence condition.

code-address2 (address formula)

Specify the address 2 of code that serves as a sequence condition.

■ Description

When the condition is succeeded the order of the specified code address 1 → code address 2, this function is used to stop the program.

■ Example

```
> SET BREAK/SEQUENCE f000, ff00
```

3.4 SET BREAK (type 4)

The SET BREAK command specifies default attributes at the break point setting.

■ Debugger

Simulator		×
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	×
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		×

■ Format

SET BREAK /DEFAULT [/SOFT|/HARD]

● Command qualifiers

/DEFAULT (Omission prohibition)

Specifies attributes at the break point setting.

/HARD (default when omitted)

A hardware break is set as the default at the break point setting.

/SOFT

A software break is set as the default at the break point setting.

■ Description

The SET BREAK command specifies default attributes at the break point setting.

■ Example

```
>SET BREAK /DEFAULT /SOFT
>SHOW BREAK /DEFAULT
default : soft
```

3.5 SHOW BREAK

The SHOW BREAK command displays the break points set by the SET BREAK command.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW BREAK [breakpoint-number [, ...]]

● Parameter

breakpoint-number (default decimal number)

Specify the break point number.

● Command qualifiers

/ALL (default when omitted)

Displays all break points.

/DATAWATCH (MB2146-09/09A/09B)

Only the data watch break information is displayed.

/SEQUENCE (MB2146-09/09A/09B)

Only the sequence break information is displayed.

/DEFAULT (MB2146-08/MB2146-07)

Displays default attributes at the break point setting.

■ Description

The SHOW BREAK command displays the break points set by the SET BREAK command.

■ Example

[Simulator debugger]

```
>SHOW BREAK
no.     en/dis     address     pass-count     symbol
1       enable     0F00        1( 1)
        Control:   BREAK
        Command:   show timer
4       disable    20DE        65535 ( 1234)
        Control:   NOBREAK
        Command:   show status
```

[Emulator debugger (MB2146-09/09A/09B)]

```
>SHOW BREAK/DATAWATCH
no.     en/dis     address     symbol
1       enable     FF20        main
>SHOW BREAK/SEQUENCE
no.     en/dis     address     level     symbol
1       enable     F000        LEVEL1    main
2       enable     FF00        LEVEL2    func
```

[Emulator debugger (MB2146-08/MB2146-07)]

```
>SHOW BREAK/DEFAULT
default : soft
>SHOW BREAK
no.     en/dis     address     kind      symbol
1       enable     F000        hard      main
2       enable     FF00        soft      func
```

3.6 CANCEL BREAK

The CANCEL BREAK command cancels a break point at the specified break point number.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

CANCEL BREAK [breakpoint-number [, ...]]

● Parameter

breakpoint-number (default decimal number)

Specify the break point number.

See the break point number by using SHOW DATABREAK command.

● Command qualifiers

/ALL

Cancels all break points.

/DATAWATCH (MB2146-09/09A/09B)

The data watch break is cancelled.

/SEQUENCE (MB2146-09/09A/09B)

Only the sequence break is cancelled.

■ Description

The CANCEL BREAK command cancels the specified break point(s).

■ Example

```
>CANCEL BREAK 1
>CANCEL BREAK 3
```

[MB2146-09/09A/09B]

```
>CANCEL BREAK / DATAWATCH
>CANCEL BREAK / SEQUENCE
```

3.7 ENABLE BREAK

The ENABLE BREAK command enables the specified break point(s).

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

ENABLE BREAK [breakpoint-number [, ...]]

● Parameter

breakpoint-number (default decimal number)

Specify the break point number.

See the break point number by using SHOW DATABREAK command.

● Command qualifiers

/ALL

Enables all break points.

/DATAWATCH (MB2146-09/09A/09B)

The data watch break is enabled.

/SEQUENCE (MB2146-09/09A/09B)

Only the sequence break is enabled.

■ Description

The ENABLE BREAK command enables the specified break point(s).

■ Example

```
>ENABLE BREAK 2
>ENABLE BREAK 3, 4
```

[MB2146-09/09A/09B]

```
>ENABLE BREAK / DATAWATCH
>ENABLE BREAK / SEQUENCE
```

3.8 DISABLE BREAK

The DISABLE BREAK command disables the specified break point(s).

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

DISABLE BREAK [breakpoint-number [, ...]]

● Parameter

breakpoint-number (default decimal number)

Specify the break point number.

See the break point number by using SHOW DATABREAK command.

● Command qualifiers

/ALL

Disables all break points.

/DATAWATCH (MB2146-09/09A/09B)

The data watch break is disabled.

/SEQUENCE (MB2146-09/09A/09B)

Only the sequence break is disabled.

■ Description

The DISABLE BREAK command disables the specified break point(s).

■ Example

```
>DISABLE BREAK 2
>DISABLE BREAK 3, 4
```

[MB2146-09/09A/09B]

```
>DISABLE BREAK / DATAWATCH
>DISABLE BREAK / SEQUENCE
```

3.9 SET DATABREAK (type 1)

The SET DATABREAK command breaks the program when data at the specified address is accessed.

■ Debugger

Simulator		×
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET DATABREAK data-access-address [, d = break-data]

● Parameters

data-access-address (address formula)
Specify the address at which data access break point set.
break-data (Default hexadecimal)
Specifies the data value that is the condition for a break.

● Command qualifiers

Specifying access attribute
/READ
Breaks program when data read-accessed.
/WRITE
Breaks program when data write-accessed.

■ Description

The SET DATABREAK command breaks the program when data at the specified address is accessed.

Use a command qualifier to set a break access type.

Breaks program both when data read-accessed and when data write-accessed if the command qualifier is omitted.

The following shows the maximum number of break points.

Emulator (MB2146-09/09A/09B) 2

■ Example

```
>SET DATABREAK &checkflg
```

Notes:

- If an automatic variable in the function is specified, the current address at which the variable is stored is set as the data access address.
- To break the program when a C variable is accessed, specify "&" before the variable as the variable address.

3.10 SET DATABREAK (type 2)

The SET DATABREAK command breaks the program when data at the specified address is accessed.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET DATABREAK data-access-address [, pass-count] [, {command : command ... }]

● Parameters

data-access-address (address formula)

Specify the address at which data access break point set.

pass-count (default decimal number)

Specify the number of times break point to be hit (1 to 65535).

Program execution specifying is stopped when this number of times is reached.

If pass-count is omitted, 1 is assumed.

This function is valid when the debugger type is only a simulator debugger.

command

Specify the command list for executing when the break address is hit.

Two or more commands can be specified by using the colon.

This function is valid when the debugger type is only a simulator debugger.

● Command qualifiers

/READ

Breaks program when data read-accessed.

/WRITE

Breaks program when data write-accessed.

■ Description

The SET DATABREAK command breaks the program when data at the specified address is accessed.

Use a command qualifier to set a break access type.

Breaks program both when data read-accessed and when data write-accessed if the command qualifier is omitted.

When such a size is specified, a break occurs under the following condition.

- For the simulator debugger, a break occurs when a specified-size access is made to the specified address.
- For the emulator debugger, an access is made from the specified address to a specified-size area.

The pass count value is set again each time the program is executed.

The following shows the maximum number of break points.

Emulator Debugger (MB2141)	65535
Simulator Debugger(Normal)	65535
Simulator Debugger(Fast)	67

■ Example

[Simulator debugger]

```
>SET DATABREAK &checkflg, 3, {SHOW TRACE:SHOW TIMER}
```

[Emulator debugger (MB2141)]

```
>SET DATABREAK &checkflg
```

Notes:

- If an automatic variable in the function is specified, the current address at which the variable is stored is set as the data access address.
- To break the program when a C variable is accessed, specify "&" before the variable as the variable address.
- The following execution type command cannot be specified for the command list.
  - GO
  - STEP
  - CALL

3.11 SHOW DATABREAK

The SHOW DATABREAK command displays the data access break point(s) set by the SET DATABREAK command.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW DATABREAK [breakpoint-number [, ...]]

● Parameter

breakpoint-number (default decimal number)

Specify the break point number.

● Command qualifier

/ALL (default when omitted)

Displays all data break points.

■ Description

The SHOW DATABREAK command displays the data access break point(s) set by the SET DATABREAK command.

■ Example

```
>SHOW DATABREAK
no.    en/dis    address    read/write    pass-count    symbol
1      enable    2000       read only     1( 0)         \trac
2      disable   2052       write only    65535(2345)
```

3.12 CANCEL DATABREAK

The CANCEL DATABREAK command cancels the specified data access break point(s).

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CANCEL DATABREAK [breakpoint-number [, ...]]

● Parameter

breakpoint-number (default decimal number)

Specify the break point number.

Use the SHOW DATABREAK command to reference the set break point numbers.

● Command qualifier

/ALL

Cancels all data access break points.

■ Description

The CANCEL DATABREAK command cancels the specified data access break point(s).

■ Example

```
>CANCEL DATABREAK 1
>CANCEL DATABREAK 1,2
```

3.13 ENABLE DATABREAK

The ENABLE DATABREAK command enables the specified data access break point(s).

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

ENABLE DATABREAK [breakpoint-number [, ...]]

● Parameter

breakpoint-number (default decimal number)

Specify the break point number.

Use the SHOW DATABREAK command to reference the set break point numbers.

● Command qualifier

/ALL

Enables all data break points.

■ Description

The ENABLE DATABREAK command enables the specified data access break point(s).

■ Example

```
>ENABLE DATABREAK 1
>ENABLE DATABREAK 1, 2
```

3.14 DISABLE DATABREAK

The DISABLE DATABREAK command disables the specified data access break point(s).

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

DISABLE DATABREAK [breakpoint-number [, ...]]

● Parameter

breakpoint-number (default decimal number)

Specify the break point number.

See the break point number by using SHOW DATABREAK command.

● Command qualifier

/ALL

Disables all data break points.

■ Description

The DISABLE DATABREAK command disables the specified data access break point(s).

■ Example

```
>DISABLE DATABREAK 1
>DISABLE DATABREAK 1, 2
```

3.15 SET EVENT

The SET EVENT command sets the event that triggers a sequencer, multitrace, and performance.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET EVENT event-number, address [& = mask] [, [!] d = data [& = mask]]

[, e = external-probe-data [& = mask]] [,p = pass-count]

● Parameters

event-number

Specify the event number (1 to 8).

address [& = mask] (address formula, data formula)

Specify a memory location taken as an event generating condition. If a mask is specified, only one portion where the bit of the mask is 1 will be valid and the others will be ignored.

If mask data is omitted, all the bits will be valid.

Automatic variables in C language cannot be specified.

d = data [& = mask] (data formula, data formula)

Specify the data taken as an event generating condition. If a mask is specified, only one portion where the bit of the mask is 1 will be valid and the others will be ignored.

If mask data is omitted, all the bits will be valid.

If ! is specified, the specified data will be assumed to be "not".

e = external-probe-data [& = mask] (data formula, data formula)

Specify the external probe data (8-bits length) taken as an event generating condition. If a mask is specified, only one portion where the bit of the mask is 1 will be valid and the others will be ignored.

If mask data is omitted, all the bits will be valid.

p = pass-count (default decimal number)

Specify the number of times events generated (1 to 255).

When pass-count is omitted the number of times of events generated is set to 1.

● Command qualifiers

- Access attributes

/CODE

Takes code access to specified address as event generating condition.

/READ

Takes read access to specified address as event generating condition.

/WRITE

Takes write access to specified address as event generating condition.

/MODIFY

Takes changing of data at specified address as event generating condition.

/MODIFY cannot be specified together with other qualifiers for specifying access attributes. If /MODIFY is given, the address mask will be disabled.

/CODE and /WRITE cannot be specified. When /CODE and /WRITE is omitted, /CODE is assumed to be specified.

■ Description

The SET EVENT command sets the event that triggers a sequencer, multitrace, and performance. If data and external probe data are omitted, they will be all ignored.

If an address is assumed to be an event condition, it will be affected by a prefetch by the MCU. Setting should be performed considering the prefetch by the MCU.

The NATIVE mode has the following restrictions:

- Only /CODE can be specified. If other access attributes are specified, an error is assumed.
- No data specifying (d =) is allowed in parameters.

The event is set in each mode set by the SET MODE command. Up to 8 events can be specified in each mode and information about them is independent in each mode. If a mode is changed, event information in a mode before the change will be saved and event information previously set in a mode after the change will be restored.

In the normal mode, the event will trigger a sequencer. To avoid this, only an event should be set.

In the MULTITRACE mode, all the set events will trigger a multitrace.

In the PERFORMANCE mode, each event number has the following meaning.

Event 1	Starting event in section 1
Event 2	Ending event in section 1
Event 3	Starting event in section 2
Event 4	Ending event in section 2
Event 5	Starting event in section 3
Event 6	Ending event in section 3
Event 7	Starting event in section 4
Event 8	Ending event in section 4

Use of the SET RUNMODE command will clear all the event settings.

■ Example

```
>SET EVENT /READ1, func1
>SET EVENT /WRITE 2, &data[2],!d=h'10
```

3.16 SHOW EVENT

The SHOW EVENT command shows the contents set by the SET EVENT command.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW EVENT [event-number [, ...]]

● Parameter

event-number

Specify the event number (1 to 8).

● Command qualifier

/ALL (default when omitted)

Shows all events.

■ Description

Displays the event settings of the specified number.

■ Example

```
>SHOW EVENT
```

3.17 CANCEL EVENT

The CANCEL EVENT command cancels the event corresponding to a specified event number.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CANCEL EVENT [event-number [, ...]]

● Parameter

event-number

Specify the event number (1 to 8).

● Command qualifier

/ALL (default when omitted)

Specifies all events.

■ Description

The CANCEL EVENT command cancels the event corresponding to a specified event number.

■ Example

```
>CANCEL EVENT
```

3.18 ENABLE EVENT

The ENABLE EVENT command enables the event temporary disabled.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

ENABLE EVENT [event-number [, ...]]

● Parameter

event-number

Specify the event number (1 to 8).

● Command qualifier

/ALL (default when omitted)

Specifies all events.

■ Description

The ENABLE EVENT command enables the event temporary disabled.

■ Example

```
>ENABLE EVENT
```

3.19 DISABLE EVENT

The DISABLE EVENT command temporarily disables the event.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

DISABLE EVENT [event-number [, ...]]

● Parameter

event-number

Specify the event number (1 to 8).

● Command qualifier

/ALL (default when omitted)

Specifies all events.

■ Description

The DISABLE EVENT command temporarily disables the event.

■ Example

```
>DISABLE EVENT
```

3.20 SET SEQUENCE

The SET SEQUENCE command sets a sequencer.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

<Format 1 >

SET SEQUENCE [/EVENT] level-number, event-number [, pass-count]

[, J = branch-level-number]

<Format 2 >

SET SEQUENCE /TIMER level-number, waiting-time [, J = branch-level-number]

<Format 3 >

SET SEQUENCE /LATCH latch-number, branch-source-level-number

branch-destination level number

● Command qualifiers classified by function

/EVENT (Default)

Sets event number to be set as trigger factor, value of pass-count, and branch-level-number where hit found.

/TIMER

Sets waiting time.

This qualifier is specified in Format 2.

/LATCH

Latches value of emulation timer as branch made from level specified in branch-level-number to another level.

The latch function permits setting of up to two conditions. This qualifier is specified in Format 3.

● Parameters

level-number (default decimal number)

Specify the level number (D'1 to D'8) to be set.

event-number (default decimal number)

Specify the event number (D'1 to D'8) to be set as a trigger factor.

pass-count (default decimal number)

Specify the pass count (D'1 to D'16777215) of the event specified as a condition. When this parameter is omitted, 1 is assumed.

waiting-time (default decimal number)

Sets the waiting time (D'1 to D'16777215) after entering the level. Elapse of the set waiting time will be a trigger. The unit is set to either 1 μs or 100 ns by the SET TIMERSCALE command.

level-number (default decimal number)

Specify the identification number (1 or 2) of the latch function.

branch-level-number (default decimal number)

Specify the level number to which a branch is made when a condition holds. If this parameter is omitted, the next level will be set. If the level number is given 0, the end is assumed to be specified (delay counter starts).

branch-source-level-number (default decimal number)

Specify the level-number indicating the branch source and branch destination of a condition for starting the latch function.

● Command qualifiers

/ENABLETRACE (default when omitted)

Enables trace sampling.

This qualifier can be used only when /EVENT or /TIMER is specified.

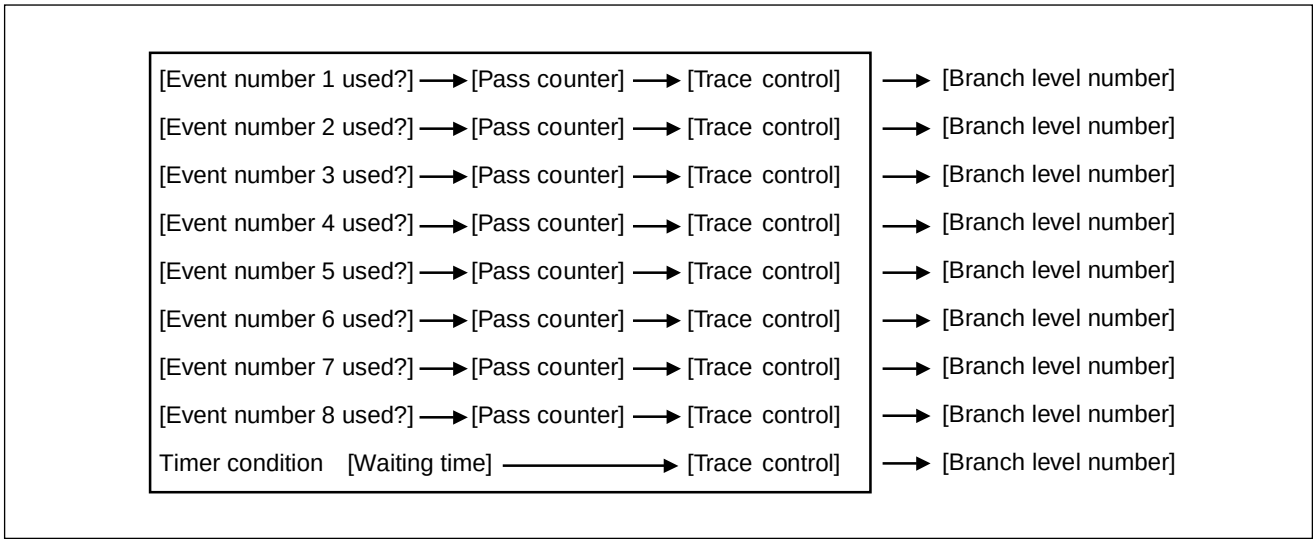
/DISABLETRACE (default when omitted)

Disables trace sampling.

This qualifier can be used only when /EVENT or /TIMER is specified.

■ Description

The SET SEQUENCE command sets a sequencer. The sequencer has 8 levels, each of which is composed as follows:


[Event number 1 used?] → [Pass counter] → [Trace control] → [Branch level number]
[Event number 2 used?] → [Pass counter] → [Trace control] → [Branch level number]
[Event number 3 used?] → [Pass counter] → [Trace control] → [Branch level number]
[Event number 4 used?] → [Pass counter] → [Trace control] → [Branch level number]
[Event number 5 used?] → [Pass counter] → [Trace control] → [Branch level number]
[Event number 6 used?] → [Pass counter] → [Trace control] → [Branch level number]
[Event number 7 used?] → [Pass counter] → [Trace control] → [Branch level number]
[Event number 8 used?] → [Pass counter] → [Trace control] → [Branch level number]
Timer condition [Waiting time] → [Trace control] → [Branch level number]

This command also sets the latch function of an emulation timer as a branch is made from a specified level to another level.

■ Example

```
>SET SEQUENCE/ON
```

3.21 SHOW SEQUENCE

The SHOW SEQUENCE command displays sequencer settings. There are following way to display them.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

<Format 1 >

SHOW SEQUENCE [/LEVEL] [level-number]

<Format 2 >

SHOW SEQUENCE /CURRENT

<Format 3 >

SHOW SEQUENCE /LATCH

● Command qualifiers classified by function

The functions of parameters can be changed by specifying following command qualifiers.

/LEVEL

Displays each level settings.

/CURRENT

When the program is executing, displays currently executing sequencer level number.

When the program is break, displays the last executed level number and pass counts of each event.

/LATCH

Displays latched timer value. When both of latch 1 and 2 are set, also displays difference of them.

● Parameter

level-number (default decimal number)

Specifies level number of sequencer. (D'1 to D'8)

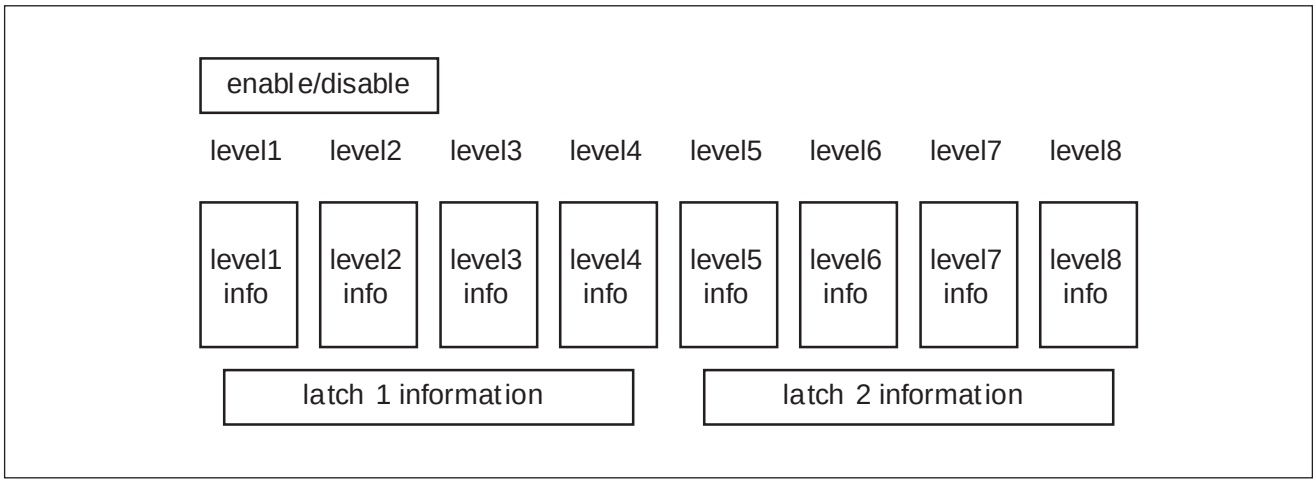
■ Description

The SHOW SEQUENCE command displays sequencer settings. There are following way to display them.

1. Global (format 1)

   When level number is omitted on format 1, displays all of level settings.

   Displays as follows. For more details, see "2.2.5.1 Setting Sequencer" in SOFTUNE Workbench User's Manual.



2. Detail (format 1)

   When level number is specified on format 1, displays detailed information of the specified level. If the event specified by the level is not set, does not display it.

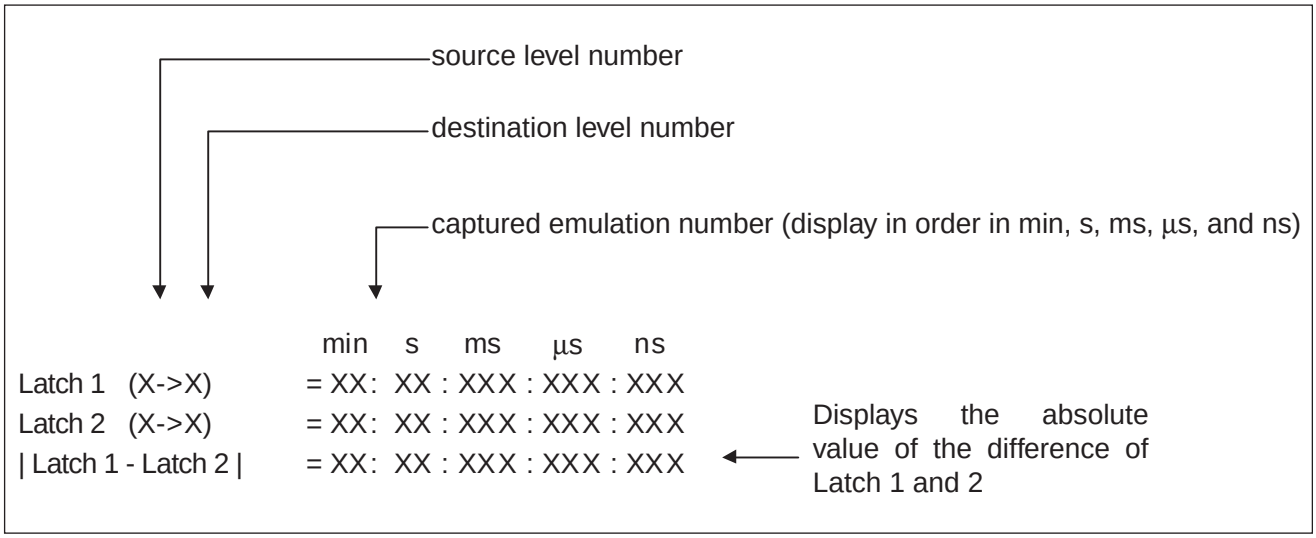
3. /CURRENT (format 2)

   When the program is executing, displays currently executing level number. Does not display other information.

   When the program is breaking, displays the last executed level number and its event information.

4. /LATCH (format 3)

   Displays latched timer values as follows.


source level number
destination level number
captured emulation number (display in order in min, s, ms, µs, and ns)
min s ms µs ns
Latch 1 (X->X) = XX: XX : XXX : XXX : XXX
Latch 2 (X->X) = XX: XX : XXX : XXX : XXX
| Latch 1 - Latch 2 | = XX: XX : XXX : XXX : XXX
Displays the absolute value of the difference of Latch 1 and 2

■ Example

```
>SHOW SEQUENCE
Sequencer is enable
   level1  level2  level3  level4  level5  level6  level7  level8
1  |1|->2  | |     | |     | |     | |     | |     | |     | |
2  | |     |2|#>3  | |     | |     | |     | |     | |     | |
3  | |     | |     |3|->end| |     | |     | |     | |     | |
4  |4|->4  | |     | |     | |     | |     | |     | |     | |
5  | |     | |     | |     |5|->1  | |     | |     | |     | |
6  | |     | |     | |     |6|#>7  | |     | |     | |     | |
7  | |     | |     | |     | |     | |     | |     |7|->end| |
8  | |     | |     | |     | |     | |     | |     |8|->4  | |
T  | |     | |     | |     | |     | |     | |     | |     | |
Latch1(2->3)=0:01:37:078.582.0  Latch2(4->7)=0:00:00:862.405.0

>SHOW SEQUENCE 1
level no. = 1
event     pass-count     trace-cntl     jmp-level
1         1              enable         2
4         1              enable         4

>SHOW SEQUENCE/LATCH
                        min   s   ms   μs   ns
Latch 1(2->3)       =   0:  01:  37:  078. 582.0
Latch 2(4->7)       =   0:  00:  00:  862. 405.0
|Latch1 - Latch2|   =   0:  01:  36:  216. 177.0
```

3.22 CANCEL SEQUENCE

The CANCEL SEQUENCE command cancels a sequencer.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

<Format 1 >

CANCEL SEQUENCE [/EVENT] level-number [, event-number]

<Format 2 >

CANCEL SEQUENCE /TIMER level-number

<Format 3 >

CANCEL SEQUENCE /LATCH latch-number

<Format 4 >

CANCEL SEQUENCE /ALL [level-number]

● Command qualifiers classified by function

/EVENT (Default)

Cancels setting of event number to be set as trigger.

This qualifier is used in Format 1.

/TIMER

Cancels waiting time.

This qualifier is used in Format 2.

/LATCH

Cancels latch function.

This qualifier is used in Format 3.

/ALL

Cancels all settings.

This qualifier is used in Format 4.

● Parameters

level-number (default decimal number)

Specify the level number to be cancelled.

event-number (default decimal number)

Specify the event number to be cancelled.

latch-number (default decimal number)

Specify the identification number of the latch function.

■ Description

When /EVENT is specified, the setting contents of the specified level and condition number for a sequencer are cancelled. If an event number is omitted, all the events set at the level will be cancelled.

When /TIMER is specified, the setting of the waiting time set at a specified level is cancelled.

When /LATCH is specified, the latch function corresponding to a specified latch number is cancelled.

If a level number is specified when /ALL is given, all the settings of the level will be deleted. If a level number is omitted, all the settings will be cancelled.

■ Example

```
>CANCEL SEQUENCE 1, 2
>CANCEL SEQUENCE 4
>CANCEL SEQUENCE /LATCH 1
```

3.23 ENABLE SEQUENCE

The ENABLE SEQUENCE command enables a sequencer.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

ENABLE SEQUENCE

■ Description

The ENABLE SEQUENCE command enables a sequencer.

■ Example

```
>ENABLE SEQUENCE
```

3.24 DISABLE SEQUENCE

The DISABLE SEQUENCE command disables a sequencer.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

DISABLE SEQUENCE

■ Description

The DISABLE SEQUENCE command disables a sequencer.

■ Example

```
>DISABLE SEQUENCE
```

3.25 SET DELAY

The SET DELAY command sets the delay count as a sequencer terminates and specifies whether or not to cause a break at the end of a delay count.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET DELAY [delay-count]

● Parameter

delay-count (default decimal number)

Specify the value (0 to 65535) of the delay between when a sequencer terminates and when a trace terminates.

The delay count is executed in machine cycles.

● Command qualifiers

/BREAK (default at start-up)

Specifies stopping of MCU execution when delay count terminates.

/NOBREAK

Specifies no stopping of MCU execution when delay count terminates.

■ Description

The SET DELAY command sets the delay count as a sequencer terminates and specifies whether or not to cause a break at the end of a delay count.

■ Example

```
>SET DELAY /NOBREAK 200
```

3.26 SHOW DELAY

The SHOW DELAY command displays the setting state of a delay count and the setting state of a break as a delay count terminates.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW DELAY

■ Description

The SHOW DELAY command displays the setting state of a delay count and the setting state of a break as a delay count terminates.

■ Example

```
>SHOW DELAY
```

CHAPTER 4

Program Execution Analysis Commands

This chapter describes the Program Execution Analysis commands.

4.1 SET MULTITRACE
4.2 SHOW MULTITRACE
4.3 CLEAR MULTITRACE
4.4 ENABLE MULTITRACE
4.5 DISABLE MULTITRACE
4.6 SEARCH MULTITRACE
4.7 SET PERFORMANCE
4.8 SHOW PERFORMANCE
4.9 CLEAR PERFORMANCE
4.10 SET COVERAGE
4.11 SHOW COVERAGE
4.12 CANCEL COVERAGE
4.13 CLEAR COVERAGE
4.14 ENABLE COVERAGE
4.15 DISABLE COVERAGE
4.16 SET SAMPLING
4.17 SHOW SAMPLING
4.18 SHOW CALLS
4.19 SHOW TIMER
4.20 CLEAR TIMER
4.21 SET TRACE

4.22 SHOW TRACE (type 1)
4.23 SHOW TRACE (type 2)
4.25 CLEAR TRACE
4.26 ENABLE TRACE
4.27 DISABLE TRACE
4.28 SEARCH TRACE
4.29 SET RAMMONITOR
4.30 SHOW RAMMONITOR
4.31 CANCEL RAMMONITOR
4.32 ENABLE RAMMONITOR
4.33 DISABLE RAMMONITOR

4.1 SET MULTITRACE

When in the MULTITRACE mode (when the event mode is set to MULTITRACE), the full break of a trace buffer is controlled.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET MULTITRACE

● Command qualifiers

/BREAK

Enables full break of trace buffer.

/NOBREAK (default when omitted)

Disables full break of trace buffer.

■ Description

In the MULTITRACE mode (when the event mode is set to MULTITRACE), the full break of a trace buffer is controlled.

When a mode is set (when a command qualifier specified), the trace buffer is cleared (by the CLEAR MULTITRACE command) to enable trace measurement (by the ENABLE MULTITRACE command).

When the full break of a trace buffer is enabled, program execution with the buffer full is suspended.

When a debugger is started, the full break of a trace buffer is disabled (by /NOBREAK).

This command setting is valid only when the event mode is set to MULTITRACE.

■ Example

```
>SET MULTITRACE /BREAK
```

4.2 SHOW MULTITRACE

The SHOW MULTITRACE command displays multitrace data stored in a trace buffer. It can be used only when the event mode is set to MULTITRACE.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

<Format 1 >

SHOW MULTITRACE /STATUS

<Format 2 >

SHOW MULTITRACE /GLOBAL [frame-number [.. frame-number]]

<Format 3 >

SHOW MULTITRACE [/LOCAL] [block-number [, frame-number [.. frame-number]]]

● Command qualifiers classified by function

/STATUS

Displays trace measurement conditions, enabled/disabled state of trace function, and storage status of trace buffer.

/GLOBAL

Displays trace data.

The specified frame number is assumed to be a global frame number given to a trace buffer.

/LOCAL (default)

Displays trace data.

The specified frame number is assumed to be a local frame number given to a trace buffer.

● Parameters

frame-number (default decimal number)

Specify the frame number of trace data to be displayed as a signed decimal value. If /GLOBAL is specified, the frame number will be the global number of the trace buffer. If /LOCAL is specified, the frame number will be the local number within one block.

block-number (default decimal number)

Specify the block number (1 to 2048) of trace data to be displayed.

- Command qualifiers

/ONEFRAME

Displays only one line of trace data.

/NEXT

Displays trace data from next block.

■ Description

The SHOW MULTITRACE command displays multitrace data stored in a trace buffer. It can be used only when the event mode is set to MULTITRACE.

Sampled trace data is assigned a number; a number locally given within each block is called a local number and a number given to the entire trace buffer is called a global number.

The local number 0 is assigned to trace data at the position where an event trigger occurs, and a negative local number is assigned to trace data sampled until the position is reached where execution stops. A global number is assigned to the oldest data, starting with 1.

- Displaying trace measurement conditions, enabled/disabled state of trace function, and storage status of trace buffer (when /STATUS specified)

en/dis = Disables/enables trace function

buffer full = Specifies full break of trace buffer

sampling = Displays sampling state of trace data

Whether sampling of trace data is on or ends is displayed. This display is used to check the program execution state. end is always displayed during break. If on is given, no trace data will be displayed.

block no. = Information on block numbers stored in trace buffer

frame no. = Information on frame numbers (global) saved in trace buffer

The minimum and maximum frame numbers are saved in the built-in variables, %TRCTOP and %TRCEND.

- Displaying trace data (when /GLOBAL and /LOCAL specified)

If a frame number is omitted, the oldest trace data or trace data in the frame next to the one last displayed is displayed.

If only a block number is specified when /LOCAL is given, all the data corresponding to the block number will be displayed. The display format is very similar to that of the SHOW TRACE machine cycle display, where all traced data are displayed in each frame. A block number is also displayed when display starts and at each portion where a block is changed.

■ Example

```
>SHOW MULTITRACE /STATUS
```

4.3 CLEAR MULTITRACE

The CLEAR MULTITRACE command clears a trace buffer.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CLEAR MULTITRACE

■ Description

The CLEAR MULTITRACE command clears a trace buffer. It is valid only when the event mode is set to MULTITRACE.

■ Example

```
>CLEAR MULTITRACE
```

4.4 ENABLE MULTITRACE

The ENABLE MULTITRACE command enables the trace function.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

ENABLE MULTITRACE

■ Description

The ENABLE MULTITRACE command enables the trace function. It is valid only when the event mode is set to MULTITRACE.

■ Example

```
>ENABLE MULTITRACE
```

4.5 DISABLE MULTITRACE

The DISABLE MULTITRACE command disables the trace function.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

DISABLE MULTITRACE

■ Description

The DISABLE MULTITRACE command disables the trace function. It is valid only when the event mode is set to MULTITRACE.

■ Example

```
>DISABLE MULTITRACE
```

4.6 SEARCH MULTITRACE

The SEARCH MULTITRACE command searches the trace data under specified conditions.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SEARCH MULTITRACE [address [&mask]] [, d= data [&mask]]

[, f = search-starting-frame-number]

● Parameters

address (address formula)

Specify the address to be searched.

data (data formula)

Specify the data to be searched.

mask (data formula)

Specify the mask-bit of address and data.

search-starting-frame-number (default decimal number)

Specify the search starting frame number (global frame number). If this parameter is omitted, a search will be made from the beginning of a trace buffer.

● Command qualifiers

- Access specifying

/CODE

Searches trace frames where code access made to specified address.

/READ

Searches trace frames where read access made to specified address.

/WRITE

Searches trace frames where write access made to specified address.

- Operation specifying

  /ALL (default when omitted)

  Searches all corresponding frames.

  /ONEFRAME

  Terminates command execution when one frame is found.

■ Description

The SEARCH MULTITRACE command searches the trace data under specified conditions. If trace data is found, it will be displayed in the same format as that in the SHOW MULTITRACE command. However, disassemble display is not executed.

If /ONEFRAME is specified, command execution will be terminated when one frame is found.

If trace data is found, the frame number will be set in the built-in variable, %FRMNUM. If multiple trace data are found, the frame number in which the last data is found will be set in the built-in variable.

■ Example

```
>SEARCH MULTITRACE 186
```

4.7 SET PERFORMANCE

The SET PERFORMANCE command sets the operation of a performance measurement buffer when it is full.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET PERFORMANCE

● Command qualifiers

/BREAK (default when omitted)

Causes break when performance measurement buffer becomes full.

/NOBREAK

Does not cause break when performance measurement buffer becomes full.

■ Description

This command determines whether causing break or not when the buffer that stores performance measuring data became full during user program execution. /BREAK can be specified to cause a break when a performance measurement buffer becomes full.

A performance measurement buffer becomes full when the event is occurred 32767 times.

This command setting is valid only when the event mode is set to PERFORMANCE.

The point at which performance is measured is set by the SET EVENT command. At performance measurement, the following items are measured:

- Time measurement

  The time between two events is measured in four sections. The starting and ending events are combined as follows:

  Section 1: Starting event 1 - Ending event 2

  Section 2: Starting event 3 - Ending event 4

  Section 3: Starting event 5 - Ending event 6

  Section 4: Starting event 7 - Ending event 8

- Counting event occurrences

  The time an event occurs is counted.

■ Example

```
>SET PERFORMANCE /BREAK
```

4.8 SHOW PERFORMANCE

The SHOW PERFORMANCE command displays the state of the setting performance, and the measured results.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

<Format 1 >

SHOW PERFORMANCE/STATUS

<Format 2 >

SHOW PERFORMANCE/COUNT [event-number]

<Format 3 >

SHOW PERFORMANCE/TIME event-number [, lower-limit, upper-limit, display-interval]

● Command qualifiers classified by function

/STATUS

Displays operation setting state of performance measurement when buffer is full.

/COUNT

Displays count result of times that event occurs.

/TIME (default when omitted)

Displays result of measuring time between two events.

● Parameters

event-number

Specify the event number (1 to 8) for displaying the setting contents.

When displaying the result of time measurement, the operation will be the same, even if the number of either the starting or ending event in the measurement section is specified.

lower-limit (default decimal number)

Specify the lower limit at which the measured time is displayed graphically. The unit is 1 μs if the minimum measured time of a timer is 1 μs, and 100 ns if it is 100 ns. If this parameter is omitted, an appropriate value will be given.

upper-limit (default decimal number)

Specify the upper limit at which the measured time is displayed graphically. The unit is 1 μs if the minimum measured time of a timer is 1 μs, and 100 ns if it is 100 ns. If this parameter is omitted, an appropriate value will be given.

display-interval (default decimal number)

Specify the interval at which the measured time is displayed graphically. The unit is 1 μs if the minimum measured time of a timer is 1 μs, and 100 ns if it is 100 ns. If this parameter is omitted, an appropriate value will be given.

■ Description

The SHOW PERFORMANCE command displays the state of performance settings.

● When /STATUS specified

The SHOW PERFORMANCE command displays the setting state of the operation of a performance measurement buffer when it is full.

● When /COUNT specified

The SHOW PERFORMANCE command displays the result of counting the time an event occurs.

● When /TIME specified

The SHOW PERFORMANCE command displays the result of time measurement. The upper limit, lower limit, and interval at which the measured time is displayed graphically can be specified.

■ Example

```
>SHOW PERFORMANCE /COUNT
```

4.9 CLEAR PERFORMANCE

The CLEAR PERFORMANCE command clears the performance measurement values.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CLEAR PERFORMANCE

■ Description

The CLEAR PERFORMANCE command clears the performance measurement values.

■ Example

```
>CLEAR PERFORMANCE
```

4.10 SET COVERAGE

The SET COVERAGE command specifies the coverage measurement area. Up to 32 areas can be set.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET COVERAGE [measurement-range]

● Parameter

measurement-range (address formula)

Specify the coverage measurement area. If /AUTOMATIC is given in the command qualifier, this parameter cannot be specified.

● Command qualifier

/AUTOMATIC (default when omitted)

Automatically sets code area of currently-loaded module.

The library area of the C compiler is not set.

■ Description

The SET COVERAGE command specifies the coverage measurement area. Up to 32 areas can be set.

The coverage area can be set within the area set as the debug area.

■ Example

```
>SET COVERAGE FE00..FFFF
```

4.11 SHOW COVERAGE

The SHOW COVERAGE command displays the result of coverage measurement in a specified measurement range.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

<Format 1>

SHOW COVERAGE [/STATUS]

<Format 2>

SHOW COVERAGE [/GENERAL] [measurement-range]

<Format 3>

SHOW COVERAGE [/TOTAL | /DETAIL] [measurement-range]

<Format 4>

SHOW COVERAGE /SOURCE [{[file-name] line-number [..line-number] | {address | address-range}}]

<Format 5>

SHOW COVERAGE /INSTRUCTION [{address | address-range}]

<Format 6>

SHOW COVERAGE /MODULE [{source-file-name | coverage-range} [, number-of-columns]]

● Parameters

measurement-range (address formula)

Specify the coverage measurement area.

If this parameter is omitted, this command displays the area in order, from the first.

file-name

Specifies a name of source file to display a coverage measurement result.

When the file name is omitted, the previously-specified file name is used.

line-number

Specifies a line number of source to display a coverage measurement result.

"$" must proceed a line number.

When line number is delimited be "..", the source within the specified range is displayed.

When the end line number is not specified, the result is displayed by 19 lines.

address (address formula)

Specifies a memory location of a code attribute.

Specifies this parameter to display a coverage measurement result corresponding to the memory location.

When a function is specified, the coverage measurement result within the address range of the function is displayed.

When other than a function is specified, the result is displayed by 19 lines.

address-range (address formula)

Specifies a memory area of a code attribute.

Specifies this parameter to display a coverage measurement result corresponding to the memory location.

source-file-name

Specifies a name of source file to display a coverage rate.

If this parameter is omitted, the coverage rate of entire load module is displayed.

coverage-range

Specifies a range of coverage rate to be displayed.

If this parameter is omitted, the entire range is displayed.

number-of-columns

Specifies a column position for a coverage rate (number of characters from the beginning of a line).

When this parameter is omitted, the number of columns is 40.

● Command qualifiers

/STATUS

Displays enabled/disabled state of coverage measurement function and coverage measurement area.

/TOTAL

Displays coverage rate in entire specified measurement range.

/GENERAL (default when omitted)

Displays result of coverage measurement in 16 addresses.

/DETAIL

Displays result of coverage measurement in an addresses.

/SOURCE

Displays a coverage measurement result in source lines.

/INSTRUCTION

Displays a coverage measurement result in machine instructions.

When SET SOURCE is set to the mode to add a source line and the memory location corresponds to the source line, this command also displays that source line.

/MODULE

Displays the coverage rate of the load module.

■ Description

The SHOW COVERAGE command displays the result of coverage measurement in a specified measurement range. If a command qualifier is omitted, the operation assumes a previous qualifier is specified.

If /GENERAL is specified, the access count will be displayed as follows:

. : No access
1 to F : Count of addresses accessed out of 16 addresses
* : 16 addresses accessed

If /DETAIL is specified, the access count will be displayed as follows:

. : No access
- : Accessed

/SOURCE,/INSTRUCTION is specified, the access count will be displayed as follows:

. : No access
* : Accessed
Blank : Not generate the code or line outside measurement range

The coverage rate outside all the coverage measurement range is displayed as "(--%)", if specify / MODULE. The asterisk "*" is displayed next the coverage rate, when a part of the area is outside the coverage measurement range.

■ Example

```
>SHOW COVERAGE

(HEX)     0X0               +1X0              +2X0
          ├─────────────────┼─────────────────┼──┤       ┌──────────┐
address   0123456789ABCDEF0123456789ABCDEF0123456      .....   ABCDEF    C0(%)
FF00      **3*F*.......                                          32.0


          >SHOW COVERAGE/SOURCE
           *   70: {
               71:    int   i;
               72:    struct table *value[16];
               73:
           *   74:    for (i=0; i<16; i++)
           *   75:        value[i] = &target[i];
               76:
           *   77:    sort_val(value, 16L);
           .   78: }
          >SHOW COVERAGE/MODULE
          sample.abs.........(84.03%)
           +- startup.asm.....(90.43%)
           +- sample.c........(95.17%)
           +- samp.c..........(100.00%)
```

Note:

At analyzing, not that the code coverage is measured by the prefetch affecting in MB2141 emulator.

4.12 CANCEL COVERAGE

The CANCEL COVERAGE cancels the coverage measurement area and disables the coverage measurement function.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CANCEL COVERAGE [measurement-range]

● Parameter

measurement-range (address formula)

Specify the measurement range to be deleted.

● Command qualifier

/ALL

Deletes all coverage measurement areas.

■ Description

The CANCEL COVERAGE cancels the coverage measurement area and disables the coverage measurement function.

■ Example

```
>CANCEL COVERAGE /ALL
```

4.13 CLEAR COVERAGE

The CLEAR COVERAGE clears a coverage buffer.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CLEAR COVERAGE

■ Description

The CLEAR COVERAGE clears a coverage buffer.

■ Example

```
>CLEAR COVERAGE
```

4.14 ENABLE COVERAGE

The ENABLE COVERAGE enables the coverage measurement function.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

ENABLE COVERAGE

■ Description

The ENABLE COVERAGE enables the coverage measurement function.

■ Example

```
>ENABLE COVERAGE
```

4.15 DISABLE COVERAGE

The DISABLE COVERAGE disables the coverage measurement function.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

DISABLE COVERAGE

■ Description

The DISABLE COVERAGE disables the coverage measurement function.

■ Example

```
>DISABLE COVERAGE
```

4.16 SET SAMPLING

The SET SAMPLING command sets the timing of data sampling according to the state of external probes.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET SAMPLING

● Command qualifiers

/INTERNAL (default at start-up)

Samples data at rising edge of machine clock.

/RISING

Samples data at rising edge of external clock input.

/FALLING

Samples data at falling edge of external clock input.

■ Description

The SET SAMPLING command sets the timing of data sampling according to the state of external probes. At startup, /INTERNAL is specified.

■ Example

```
>SET SAMPLING /INTERNAL
```

4.17 SHOW SAMPLING

The SHOW SAMPLING displays the state of channels for external probes and the timing of data sampling.

■ Debugger

Simulator		×
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW SAMPLING

■ Description

The SHOW SAMPLING displays the state of channels for external probes and the timing of data sampling.

■ Example

```
>SHOW SAMPLING
```

4.18 SHOW CALLS

The SHOW CALLS command displays the calling history until current function.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SHOW CALLS [call-frame-count]

● Parameter

call-frame-count (default decimal number)

Specifies count of call frames requiring information (D'1 to D'256).

■ Description

The SHOW CALLS command displays the calling history until current function.

When call-frame-count is not specified, the command displays up to 256 frames.

When the function to be displayed contains an argument, the command displays the argument as a hexadecimal number.

If there is no C language debug information, the command displays the function address instead of the function name.

The command analyzes accumulated stack data and determines which data to display according to the analysis result. It analyzes accumulated stack data according to the stack format used when C language calls the function.

Note the following when using the SHOW CALLS command:

- The command cannot be used in the programs coded in assembler.
- In the optimized program, the command may be unable to display data normally.
- If the program is not compiled with debug information, the command displays the address instead of the function name. However, if the program breaks at the beginning of the function, the command cannot display data normally.

■ Example

```
>SHOW CALLS
checker (12, 8)
main (3, 4)
```

4.19 SHOW TIMER

The SHOW TIMER command displays the instruction execution cycle count, step count, and time of the executed program in decimal notation.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○*
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

*: This command is only available if the BGM adapter (MB2146-09A or later) and MCU board (version 02B or later) are used.

■ Format

SHOW TIMER

■ Description

The SHOW TIMER command displays the instruction execution cycle count, step count, and time of the executed program in decimal notation.

The numeric values displayed are those after the RESET command has been executed and those when and after program execution is started by the preceding GO, STEP, or CALL command.

The contents of the display are different in each debugger. For details, refer to the following sections of "SOFTUNE WORKBENCH USER'S MANUAL".

- Simulator debugger : "2.1.9 Measuring Execution Cycle Count"
- Emulator debugger (MB2141) : "2.2.10 Measuring Execution Time"
- Emulator debugger (MB2146-09/09A/09B) : "2.3.8 Measuring Execution Cycle Count"

■ Example

```
>SHOW TIMER
```

Simulator Debugger

```
<cycle>     From initialize:        6410[Cycle]
            From Last Executed:     415[Cycle]
<step>      From initialize:        1251[Step]
            From Last Executed:     121[Step]
```

Emulator Debugger (MB2141)

```
<timer>     From initialize:        0h00m42s108ms264μs[Time]
            From Last Executed:     0h00m03s623ms874μs[Time]
```

Note:

This command is available under certain conditions when using the emulator debugger for the MB2146-09. For details, refer to "2.3 Emulator Debugger (MB2146-09/09A/09B)" of "SOFTUNE Workbench User's Manual".

4.20 CLEAR TIMER

The CLEAR TIMER command initializes the timer measurement execution result.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○*
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

*: This command is only available if the BGM adapter (MB2146-09A or later) and MCU board (version 02B or later) are used.

■ Format

CLEAR TIMER

■ Description

The CLEAR TIMER command initializes executed number of instruction execution cycles, number of steps of programs, and the result of a measurement at time are.

■ Example

```
>CLEAR TIMER
```

Note:

This command is available under certain conditions when using the emulator debugger for the MB2146-09. For details, refer to "2.3 Emulator Debugger (MB2146-09/09A/09B)" of "SOFTUNE Workbench User's Manual".

4.21 SET TRACE

The SET TRACE command controls the trace buffer full break.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SET TRACE

● Command qualifiers

/BREAK

Enables trace buffer-full break.

/NOBREAK (default at start-up)

Disables trace buffer-full break.

■ Description

Enabling the trace buffer-full break, suspends program execution when the trace buffer becomes full.

■ Example

```
>SET TRACE/BREAK
```

4.22 SHOW TRACE (type 1)

The SHOW TRACE command displays the trace data stored in the trace buffer.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW TRACE [/DATA] [trace-number [.. trace-number]]

● Command qualifier classified by function

/DATA (default when omitted)

Displays traced data.

● Parameter

trace-number (default decimal number)

Specify the number of trace data to be displayed with decimal number.

● Command qualifiers

/CYCLE

Displays the trace data in valid bus cycle.

When the debugger type is only the emulator debugger (MB2141), this command qualifier can be specified.

/INSTRUCTION(default when omitted)

Displays the trace data by the disassemble form.

/SOURCE

Displays the trace data by the source lines form.

/ONEFRAME

Displays trace data only by one line.

/NEXT

Displays from the frame which next level of sequencer being traced. This qualifier can be specified only Emulator (MB2141).

/FILE

Save trace data to a file.

/APPEND

Add and save trace data to a file.

Effective when"/FILE" is specified.

■ Description

The SHOW TRACE command displays the trace data stored in the trace buffer.

Sampled trace data is assigned numbers. Trace data in the execution stop location (trigger point) is assigned number 0. The sampled trace data is assigned negative numbers until the execution stop location is reached. These numbers are called frame numbers.

When /APPEND is specified, trace data is added and saved.

■ Example

```
> SHOW TRACE/SOURCE -300
frame no.    source
-00278:      Sample.c$39                          break
-00275:      Sample.c$43           tblp [i -1] = p;
-00237:      Sample.c$44      }
-00234:      Sample.c$30      while (max > 1) {
-00217:      Sample.c$31           p = tblp [max - 1]
-00182:      Sample.c$32           tblp [max - 1] = tblp {0};
-00133:      Sample.c$33           max--;
-00120:      Sample.c$34           i = 1;
-00112:      Sample.c$35           while ((j = 2*i) < = max) {
-00079:      Sample.c$43      tblp [i - 1] = p;
-00041:      Sample.c$44      }
-00038:      Sample.c$30      while (max > 1) {
> SHOW TRACE/FILE/APPEND     C:\sample.log
```

4.23 SHOW TRACE (type 2)

The SHOW TRACE command displays the trace conditions.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW TRACE /STATUS

● Command qualifier classified by function

/STATUS

Displays trace measurement conditions, enabled/disabled state of trace function, and storage status of trace buffer.

■ Description

The SHOW TRACE command displays the trace conditions. Displays trace measurement conditions, enabled/disabled state of trace function, and storage status of trace buffer.

■ Example

```
>SHOW TRACE/STATUS
en/dis        =   enable
buffer full   =   nobreak
sampling      =   end
frame no.     =   -22639 to 00000
Step no.      =   -22639 to 00000
```

4.24 SHOW TRACE (type 3)

The SHOW TRACE command preserves the trace data stored in the trace buffer in the file.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

SHOW TRACE /FILE file name

● Parameter

file name

Specify a file name with full path to which the trace data will be saved.
When the extension is omitted, .log is used as the extension.

● Command qualifier classified by function

- Specification of the method preserved

/FILE

Preserves the trace data in the file.

● Command qualifiers

- Specification of the method preserved

/APPEND

Preserves the trace data in the file additionally. /APPEND is effective only when /FILE is specified.

- Specification of data preserved

/CYCLE (Default when omitted when MCU operational mode is native mode)

Preserves the trace data at the effective bus cycle.

/INSTRUCTION (Default when omitted when MCU operational mode is debug mode)

Preserves the trace data by the disassemble form.

/SOURCE

Preserves the trace data by each source line.

■ Description

The SHOW TRACE command preserves the trace data stored in the trace buffer in the file.

Data is preserved by the binary form.

■ Example

```
>SHOW TRACE/FILE trcdata
```

4.25 CLEAR TRACE

The CLEAR TRACE command clears the trace buffer.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

CLEAR TRACE

■ Description

The CLEAR TRACE command clears the trace buffer.

■ Example

```
>CLEAR TRACE
```

4.26 ENABLE TRACE

The ENABLE TRACE command enables the trace function.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

ENABLE TRACE

■ Description

The ENABLE TRACE command enables the trace function.

■ Example

```
>ENABLE TRACE
```

4.27 DISABLE TRACE

The DISABLE TRACE command disables the trace function.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

DISABLE TRACE

■ Description

The DISABLE TRACE command disables the trace function.

■ Example

```
>DISABLE TRACE
```

4.28 SEARCH TRACE

The SEARCH TRACE command searches for trace data according to the specified condition.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

<Format 1 >

SEARCH TRACE [address [& = mask-data]] [, f = search-start-number]

<Format 2 >

SEARCH TRACE [d = data [& = mask-data]] [, f= search-start-number]

● Parameters

address (address formula)

Specify the address to be searched.

data (data formula) (MB2141)

Specify the data to be searched.

mask-data (data formula)

Specify the masking and searching of address and data.

Only the bits set to 1 are to be compared for search.

search-start-number (default decimal number)

Specify the search start frame number.

When this parameter is omitted, the command starts data search from the beginning of the trace buffer.

● Command qualifiers

/ALL (default when omitted)

Searches for all associated frames.

/ONEFRAME

Terminates trace data search when one frame found.

/READ (MB2141)

Searches trace frame or step where read access made to specified address.

/WRITE (MB2141)

Searches trace frame or step where write access made to specified address.

/LEVEL (MB2141)

Searches for point where level of sequencer changes.

If this qualifier is given, parameters other than the search start number cannot be specified.

■ Description

The SEARCH TRACE command searches for trace data according to the specified condition.

When the trace data matching the condition is found, the command displays it in the same format as the SHOW TRACE command.

When /ONEFRAME is specified, the debugger terminates this command when one frame is found.

■ Example

```
>SEARCH TRACE F2E0
  frame no.   address   data   mnemonic   level   ext-probe
  -03476      :F2E0     81     CLRC       1       11111111
  -03184      :F2E0     81     CLRC       1       11111111
  -02696      :F2E0     81     CLRC       1       11111111
  -02391      :F2E0     81     CLRC       1       11111111
  -01903      :F2E0     81     CLRC       1       11111111
  -01598      :F2E0     81     CLRC       1       11111111
  -01097      :F2E0     81     CLRC       1       11111111
  -00609      :F2E0     81     CLRC       1       11111111
```

4.29 SET RAMMONITOR

The SET RAMMONITOR command sets the watching address and data size with the RAM monitor function.

■ Debugger

Simulator		×
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	○*
	(MB2146-08)	×
	(MB2146-07)	○
Monitor		×

*: This command is available if only the BGM adapter (MB2146-09B) or the MCU board (MB2146-301B-E or MB2146-303B-E) is used.

■ Format

SET RAMMONITOR address

● Parameters

address (address formula)

Specify the address to be watched.

● Command qualifiers

- Data size

/BYTE (default when omitted)

Specifies data size by 8 bits.

/WORD

Specifies data size by 16 bits.

■ Description

The SET RAMMONITOR command sets the watching address and data size with the RAM monitor function.

The maximum addresses count to be watched is as follows.

- MB2146-09B: 32 points
- MB2146-07: 16 points

For detail of the RAM monitoring function, refer to each section of "SOFTUNE Workbench user's manual".

- MB2146-09B: "2.3.7 RAM Monitoring"
- MB2146-07: "2.5.7 RAM Monitoring"

■ Example

```
> SET  RAMMONITOR/WORD  \main\data_1
> SHOW  RAMMONITOR
RAM Monitoring : enable
ch 0 : 0140 /word : \main\data_1
ch 1 : 1200 /byte
ch 2 : 1240 /byte
```

4.30 SHOW RAMMONITOR

The SHOW RAMMONITOR command displays the content set by the RAM monitor function.

■ Debugger

Simulator		×
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	◎*
	(MB2146-08)	×
	(MB2146-07)	◎
Monitor		×

*: This command is available if only the BGM adapter (MB2146-09B) or the MCU board (MB2146-301B-E or MB2146-303B-E) is used.

■ Format

SHOW RAMMONITOR

■ Description

Displays the content set by the RAM monitor function.

Set contents are displayed by the following priority.

1. The order rearranged by RAM monitor dialog
   Refer to "4.4.15 RAM Monitoring" in "SOFTUNE Workbench operation manual" for details.
2. The order specified by SET RAMMONITOR command

■ Example

```
> SHOW  RAMMONITOR
RAM Monitoring : enable
ch 0 : 0140 /word : \main\data_1
ch 1 : 1200 /byte
ch 2 : 1240 /byte
```

4.31 CANCEL RAMMONITOR

The CANCEL RAMMONITOR command cancels the address watching by the RAM monitor function.

■ Debugger

Simulator		×
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	○*
	(MB2146-08)	×
	(MB2146-07)	○
Monitor		×

*: This command is available if only the BGM adapter (MB2146-09B) or the MCU board (MB2146-301B-E or MB2146-303B-E) is used.

■ Format

```
CANCEL  RAMMONITOR [address]
CANCEL  RAMMONITOR /ALL
```

● Parameters

address (address formula)

Specify the watching address to be canceled.

● Command qualifiers

/ALL

Cancels all watching address.

■ Description

The CANCEL RAMMONITOR command cancels the address watching by the RAM monitor function.

■ Example

```
> SHOW  RAMMONITOR
RAM Monitoring : enable
ch 0 : 0140 /word : \main\data_1
ch 1 : 1200 /byte
ch 2 : 1240 /byte
> CANCEL  RAMMONITOR  0x1240
> SHOW  RAMMONITOR
RAM Monitoring : enable
ch 0 : 0140 /word : \main\data_1
ch 1 : 1200 /byte
```

4.32 ENABLE RAMMONITOR

The ENABLE RAMMONITOR command enables the RAM monitor function.

■ Debugger

Simulator		×
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	○*
	(MB2146-08)	×
	(MB2146-07)	○
Monitor		×

*: This command is available if only the BGM adapter (MB2146-09B) or the MCU board (MB2146-301B-E or MB2146-303B-E) is used.

■ Format

ENABLE RAMMONITOR

■ Description

The ENABLE RAMMONITOR command enables the RAM monitor function.

■ Example

```
> SHOW  RAMMONITOR
RAM Monitoring : disable
ch 0 : 0140 /word : \main\data_1
ch 1 : 1200 /byte
ch 2 : 1240 /byte
> ENABLE  RAMMONITOR
> SHOW  RAMMONITOR
RAM Monitoring : enable
ch 0 : 0140 /word : \main\data_1
ch 1 : 1200 /byte
ch 2 : 1240 /byte
```

4.33 DISABLE RAMMONITOR

The DISABLE RAMMONITOR command disables the RAM monitor function.

■ Debugger

Simulator		×
Emulator	(MB2141)	×
	(MB2146-09/09A/09B)	○*
	(MB2146-08)	×
	(MB2146-07)	○
Monitor		×

*: This command is available if only the BGM adapter (MB2146-09B) or the MCU board (MB2146-301BE or MB2146-303B-E) is used.

■ Format

DISABLE RAMMONITOR

■ Description

The DISABLE RAMMONITOR command disables the RAM monitor function.

■ Example

```
> SHOW  RAMMONITOR
RAM Monitoring : enable
ch 0 : 0140 /word : \main\data_1
ch 1 : 1200 /byte
ch 2 : 1240 /byte
> DISABLE  RAMMONITOR
> SHOW  RAMMONITOR
RAM Monitoring : disable
ch 0 : 0140 /word : \main\data_1
ch 1 : 1200 /byte
ch 2 : 1240 /byte
```

CHAPTER 5

Memory/Register Operation Commands

This chapter describes the Memory/Register Operation commands.

5.1 EXAMINE
5.2 ENTER
5.3 SET MEMORY
5.4 SHOW MEMORY
5.5 SEARCH MEMORY
5.6 SET REGISTER
5.7 SHOW REGISTER
5.8 COMPARE
5.9 FILL
5.10 MOVE
5.11 DUMP
5.12 COPY
5.13 VERIFY

5.1 EXAMINE

The EXAMINE command analyzes the specified formula in C language and displays the result.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

EXAMINE expression [, ...]

● Parameter

expression (address formula)

Specify the expression to be analyzed.

● Command qualifiers

/BINARY

Specifies the formula solution to be displayed as binary number.

/OCTAL

Specifies the formula solution to be displayed as octal number.

/DECIMAL

Specifies the formula solution to be displayed as decimal number.

/HEXADECIMAL

Specifies the formula solution to be displayed as hexadecimal number.

/SINGLE

Specified to display the analyzed formula solution as a single-precision floating-point number.

/DOUBLE

Specified to display the analyzed formula solution as a double-precision floating-point number.

■ Description

The EXAMINE command analyzes the specified formula in C language and displays the result.

When a variable is specified, the command displays the data.

When a variable of structure or union or class type is specified, the command displays all the member values. When only an array name is specified, the command displays all the data of that array.

When the display base number of a command qualifier is omitted, the base number specified by the SET RADIX command is assumed.

■ Example

```
>EXAMINE strsym
strsym ={
  a = H'20
  b = H'4A30
  c = H'3012
}
>EXAMINE strsym.a
strsym.a = H'20
>EXAMINE flags [0]
flags [0] = H'03
>EXAMINE flags
flags [0] = H'05
flags [1] = H'50
flags [2] = H'10
flags [3] = H'2A
>EXAMINE/DECIMAL count
count = D'12
>EXAMINE/HEXADECIMAL count
count = H'0C
>EXAMINE/DECIMAL fwork
fwork = 2.36S+1
```

Note:

For the method of specifying formula, refer to the following sections of "SOFTUNE WORKBENCH OPERATION MANUAL".

- "2.1.1 Data and Address Formulas (Numerical constant)"
- "2.1.2 Data and Address Formulas (Symbols, Line Numbers, Character Constants)"
- "2.1.3 Data and Address Formulas (Register name, Flag name)"

5.2 ENTER

The ENTER command assigns the specified data to the specified variable.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

ENTER variable = data

● Parameters

variable (address formula)

Specify the variable where data to be stored.

data (data formula)

Specify the data to be stored.

● Command qualifiers

- Data length

/BYTE

Stores specified value in specified memory location as 8-bit length.

/WORD

Stores specified value in specified memory location as 16-bit length.

/LONG

Stores specified value in specified memory location as 32-bit length.

/SINGLE

Stores specified value in specified memory location as single-precision floating-point number.

/DOUBLE

Stores specified value in specified memory location as double-precision floating-point number.

■ Description

The ENTER command assigns the specified data to the specified variable.

Specifying the type of command qualifier enables data to be assigned at the specified size.

■ Example

```
>ENTER tmcnt = 10
>ENTER work = 6A5
>ENTER tmp = F2BF
>ENTER fsymbol = F'10.55S+2
>ENTER/WORD work = 1234
```

5.3 SET MEMORY

The SET MEMORY command stores the specified data in the specified memory location by storage-address according to the type of the specified command qualifier.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SET MEMORY [storage-address] = data [, ...]

● Parameters

storage-address (address formula)

Specify the memory location where specified data to be stored.

data (data formula)

Specify the value to be stored.

● Command qualifiers

- Data length

/BIT

Stores specified value in specified memory location as bit-length data.

/BYTE (default when omitted)

Stores specified value in specified memory location as 8-bit length.

/WORD

Stores specified value in specified memory location as 16-bit length.

/LONG

Stores specified value in specified memory location as 32-bit length.

/SINGLE

Stores specified value in specified memory location as single-precision floating-point number.

/DOUBLE

Stores specified value in specified memory location as double-precision floating-point number.

/STRING

Stores value specified in character string in specified memory location as ASCII code data.

■ Description

The SET MEMORY command stores the specified data in the specified memory location by storage-address according to the type of the specified command qualifier.

When storage-address is omitted, the command stores the specified data in the memory location next to the memory location last accessed by the SHOW MEMORY or SET MEMORY commands. The type of the data to be stored is the same as that of the last accessed memory data.

When only a period (.) is specified in storage-address, the command stores the data in the memory location last accessed by the SHOW MEMORY or SET MEMORY commands.

In this case, the type of the data to be stored is also the same as that of the last accessed memory data.

If the type of command qualifier is omitted, /BYTE is assumed.

■ Example

```
>SET MEMORY/BYTE 1000 = 10
>SET MEMORY/HALFWORD 1030 = 6A5
>SET MEMORY/WORD 1050 = 1DF2BF
>SET MEMORY/STRING 2000 = "ST"
>SET MEMORY . = 45
>SET MEMORY/BIT 8000:3 = 1
>SET MEMORY/SINGLE 2050 = F'10.55S+2
```

5.4 SHOW MEMORY

The SHOW MEMORY command displays data in the memory location, specified by address or address-range, according to the type of specified data.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SHOW MEMORY [{address | address-range} [, ...]]

● Parameters

address (address formula)

Specify the address in memory location to be checked.

address-range (address formula)

Specify the memory area range to be checked.

● Command qualifiers

/BIT

Specifies that value to be checked to be displayed as 1-bit length.

/BYTE (default when omitted)

Specifies that value to be checked to be displayed as 8-bit length.

/WORD

Specifies that value to be checked to be displayed as 16-bit length.

/LONG

Specifies that value to be checked to be displayed as 32-bit length.

/SINGLE

Specifies that value to be checked to be displayed as single-precision floating-point number.

/DOUBLE

Specifies that value to be checked to be displayed as double-precision floating-point number.

/ASCII

Specifies that value to be checked to be displayed as ASCII characters.

/STRING

Specifies that value to be checked to be displayed as character string.

/BINARY

Specifies that value to be checked to be displayed as binary number.

/OCTAL

Specifies that value to be checked to be displayed as octal number.

/DECIMAL

Specifies that value to be checked to be displayed as decimal number.

/HEXADECIMAL

Specifies that value to be checked to be displayed as hexadecimal number.

■ Description

The SHOW MEMORY command displays data in the memory location, specified by address or address-range, according to the type of specified data. However, when /BIT is specified, address-range cannot be specified.

When address and address-range are omitted, the command displays data in the memory location next to the memory location last accessed by the SHOW MEMORY or SET MEMORY commands.

The type of the data to be displayed is the same as that of the last-accessed memory data.

When only a period (.) is specified in address, the command displays the data in the memory location last accessed by the SHOW MEMORY or SET MEMORY commands.

In this case, the type of data to be displayed is also the same as that of the last accessed memory data.

If the command qualifier type is omitted, /BYTE is assumed.

If the display base number of a command qualifier is omitted, the base number specified by the SET RADIX command is assumed.

■ Example

```
>SHOW MEMORY/DECIMAL C000
C000 = D'49
>SHOW MEMORY/BINARY C000
C000 = B'00110001
>SHOW MEMORY/HEXADECIMAL C000..C001
C000 = H'31
C001 = H'32
>SHOW MEMORY/HEXADECIMAL/WORD C000
C000 = H'3132
>SHOW MEMORY/HEXADECIMAL/LONG C000
C000 = H'31323334
>SHOW MEMORY/HEXADECIMAL C000,C02D
C000 = H'31
C02D = H'64
```

```
>SHOW MEMORY/ASCII C00A
C00A = 'a'
>SHOW MEMORY/SINGLE/DECIMAL C030
C030 = 2.593151S-9
>SHOW MEMORY/BYTE C000
C000 = H'31
>SHOW MEMORY
C001 = H'32
>SHOW MEMORY
C002 = H'33
```

5.5 SEARCH MEMORY

The SEARCH MEMORY command searches the specified memory for the specified data and displays the address matching the data.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SEARCH MEMORY address-range = data [, ...] [, S = skip-byte-count]

● Parameters

address-range (address formula)

Specify the memory area to be searched.

data (data formula)

Specify the data to be searched.

skip-byte-count (data formula)

Specify the number of bytes to be skipped during search.
H'1 to H'FFFF can be specified.
If this parameter is omitted, the data length is assumed.

● Command qualifiers

- Data length

/BYTE (default when omitted)

Searches for specified data as 8-bit length data.

/WORD

Searches for specified data as 16-bit length data.

/LONG

Searches for specified data as 32-bit length data.

/ASCII

Searches for specified data as ASCII character strings.

■ Description

The SEARCH MEMORY command searches the specified memory for the specified data and displays the address matching the data.

■ Example

```
>SEARCH MEMORY 2000..3000 = 88
found at = 2050
found at = 2577
found at = 2BDF
```

5.6 SET REGISTER

The SET REGISTER command sets the specified value in the specified register or flag.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SET REGISTER register-name = data

● Parameters

register-name

Specify the name of register or flag to be modified.

For register and flag names, refer to the "Appendix A List of Register Names" of "SOFTUNE Workbench Operation Manual".

data (data formula)

Specify the value to be set in specified register or flag.

■ Description

The SET REGISTER command sets the specified value in the specified register or flag.

■ Example

```
>SET REGISTER PC = 1000
>SET REGISTER C = 1
```

5.7 SHOW REGISTER

The SHOW REGISTER command displays the contents of the specified register or flag in hexadecimal notation.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SHOW REGISTER register-name

● Parameter

register-name

Specify the name of register or flag to be checked. When the register name is omitted, all registers and content of the flag are displayed.

For register and flag names, refer to the "Appendix A List of Register Names" of "SOFTUNE Workbench Operation Manual".

● Command qualifier

/ALL (default when omitted)

Displays contents of all registers and flags.

■ Description

The SHOW REGISTER command displays the contents of the specified register or flag in hexadecimal notation.

When not set, each flag register displays "-". When set, it displays the flag name.

■ Example

```
>SHOW REGISTER IP
IP = F357
>SHOW REGISTER
PC = F357   A = 00FA   T = 0001   IX = 00FA
SP = 00D6   EP = 0019   IL = 3   FLAGS = --**-Z--
RP = 00   R0 = 00   R1 = 00   R2 = FF
R3 = FF   R4 = FF   R5 = FF   R6 = FF
R7 = FF
```

5.8 COMPARE

The COMPARE command compares memory data.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		×

■ Format

COMPARE compare-origin-address-range, comparison-destination-address

● Parameters

compare-origin-address-range (address formula)

Specify the memory area of compare origin.

comparison-destination-address (address formula)

Specify the comparison destination address.

■ Description

The COMPARE command compares memory data.

When no difference is found as a result of the comparison, the COMPARE command displays "Not found".

When a difference is found, the command displays in hexadecimal notation the memory location of the compare origin to the left and the memory location of the comparison destination to the right.

■ Example

```
>COMPARE 2000..3000, 4000
address    source    destination    address
2050       35        10             4050
2051       40        00             4051
```

5.9 FILL

The FILL command fills the specified memory area with any data.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		×

■ Format

FILL address-range = data [, ...]

● Parameters

address-range (address formula)

Specify the memory range to be filled.

data (data formula)

Specify the data filling.

● Command qualifiers

- Data length

/BYTE (default when omitted)

Specifies filling the data with 8-bit length data.

/WORD

Specifies filling the data with 16-bit length data.

/LONG

Specifies filling the data with 32-bit length data.

/ASCII

Specifies filling the data with ASCII character string data.

■ Description

The FILL command fills the specified memory area with any data.

■ Example

```
>FILL 2000..2FFF = 23
```

5.10 MOVE

The MOVE command transfers data from the specified memory area to the specified transfer destination.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		×

■ Format

MOVE transfer-source-address-range, transfer-destination-address

● Parameters

transfer-source-address-range (address formula)

Specify the memory area from where data transferred.

transfer-destination-address (address formula)

Specify the memory location to where data to be transferred.

■ Description

The MOVE command transfers data from the specified memory area to the specified transfer destination.

■ Example

```
>MOVE 2000..3000, 4000
```

5.11 DUMP

The DUMP command dumps data in the specified memory area.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

DUMP [{starting-address | address-range}]

● Parameters

starting-address (address formula)

Specify the memory location address where dump to be started.

address-range (address formula)

Specify the memory area range to be dumped.

● Command qualifiers

- Display unit

/BIT

Dumps data in bits.

/BYTE (default when omitted)

Dumps data in 8 bits.

/WORD

Dumps data in 16 bits.

/LONG

Dumps data in 32 bits.

■ Description

The DUMP command dumps data in the specified memory area.

When only starting-address is specified, the DUMP command displays the first 16 lines in the output window.

When no parameter is specified, the command displays the memory location next to the memory location last-displayed as a result of previous command execution.

■ Example

```
>DUMP  100..118
address +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F  ---ascii--
0100    00 00 41 42 43 00 00 00 00 00 00 00 00 00 00 00 ..ABC......
0110    53 49 4D 55 4C 41 54 4F 52                      SIMULATOR
>
>DUMP/WORD  100..118
address   +0    +2    +4    +6    +8    +A    +C    +E   ---ascii--
0100      0000  4241  0043  0000  0000  0000  0000  0000 ..ABC.....
0110      4953  554D  414C  4F54  0052                   SIMULATOR
>
>DUMP/BIT  5
address   :7  :6  :5  :4  :3  :2  :1  :0  HEX
0005      0   1   1   0   1   0   1   0   6A
0006      1   1   1   1   0   1   0   0   F4
     ...
     ...
```

5.12 COPY

The COPY command copies data in user memory corresponding to a specified memory area to emulation memory corresponding to a specified memory area.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

COPY transfer-source-address-range

● Parameter

transfer-source-address-range

Specify the transfer-source memory area.

■ Description

The COPY command copies data in user memory corresponding to a specified memory area to emulation memory corresponding to a specified memory area. The specified memory area must be mapped as emulation memory.

■ Example

```
>COPY 6000..60FF
```

5.13 VERIFY

The VERIFY command collates data in user memory corresponding to a specified memory area with data in emulation memory.

■ Debugger

Simulator		×
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	×
	(MB2146-08)	×
	(MB2146-07)	×
Monitor		×

■ Format

VERIFY collating-address-range

● Parameter

collating-address-range

Specify the collating memory area.

■ Description

The VERIFY command collates data in user memory corresponding to a specified memory area with data in emulation memory. If there is no difference as a result of collating, the system waits for completion of command execution. The collating area must be mapped as emulation memory.

■ Example

```
>VERIFY 6000..60FF
```

CHAPTER 6

Line Assemble and Disassemble Commands

This chapter describes the Line Assemble and Disassemble commands.

6.1 ASSEMBLE

6.2 DISASSEMBLE

6.1 ASSEMBLE

The ASSEMBLE command line-assembles the entered mnemonic and operand, and stores the instruction code in the specified memory location.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

ASSEMBLE [starting-address] = assemble-character-string

● Parameters

starting-address (address formula)

Specify a starting address of memory containing line-assembled codes.

assemble-character-string (character string)

Specify a character string to be line-assembled. Please enclose the string in double quotation marks ' " ' (character).

■ Description

The ASSEMBLE command line-assembles the entered mnemonic and operand, and stores the instruction code in the specified memory location.

When starting-address is omitted, input wait of mnemonic is assumed from the memory location next to the memory location previously-stored last instruction code.

■ Example

```
>ASSEMBLE F370="MOVW a, #0000"
>ASSEMBLE F373="XCHW      A,T"
>DISASSEMBLE F370.. +8
F370     E40000     MOVW     A,#0000
F373     43         XCHW     A,T
F374     10         SWAP
F375     F908       BC       F37F
F377     10         SWAP
F378     54FF       X0R      A,#FF
```

6.2 DISASSEMBLE

The DISASSEMBLE command disassembles data in the specified memory location.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

DISASSEMBLE [{starting-address | address-range}]

● Parameters

starting-address (address formula)

Specify a starting address of memory to be disassembled.

address-range (address formula)

Specify a range of memory to be disassembled.

■ Description

The DISASSEMBLE command disassembles data in the specified memory location and displays it in the output window.

When only starting-address is specified, the command disassembles and displays data by 16 lines.

When only a period (.) is specified in parameter, the command starts disassembled data display from the address indicated by the current program counter.

When parameter are omitted, the command displays disassembled data by 16 lines, starting from the line next to the last displayed line.

When SET SOURCE command is set to the mode to add source lines and the memory location corresponds to the source line, the DISASSEMBLE command also displays the source line.

■ Example

```
>DISASSEMBLE   F3B0..F3B8
F3B0 48         MOV          R0,A
F3B1 10         SWAP
F3B2 49         MOV          R1,A
F3B3 43         XCHW         A,T
F3B4 E40000     MOVW         A,#0000
F3B7 43         XCHW         A,T
F3B8 F802       BNC          F3BC
>DISASSEMBLE
F36D C0         INCW         A
F36E D9         DEC          R1
F36F E3         MOVW         EP,A
F370 E40000     MOVW         A,#0000
F373 43         XCHW         A,T
F374 10         SWAP
F375 F908       BC           F37F
F377 10         SWAP
F378 54FF       X0R          A,#FF
F37A 10         SWAP
F37B 54FF       X0R          A,#FF
F37D C0         INCW         A
F37E C9         INC          R1
F37F E40000     MOVW         A,#0000
F382 43         XCHW         A,T
F383 8811       MOV          R0,#11
```

CHAPTER 7

Load and Save Commands

This chapter describes the Load and Save commands.

7.1 LOAD
7.2 SAVE

7.1 LOAD

The LOAD command loads the specified file.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

LOAD file-name [, address] [, file-offset [, byte-count]]

● Parameters

file-name

Specify a name of file to be loaded.
The default extension depends on the command qualifier.

address (address formula)

Specify a memory location (address) where memory image file is loaded.
This parameter is valid only when command qualifier /BINARY is specified.
Specifying other command qualifiers results in an error.

file-offset (data formula)

Specify an offset of read start data in specified file.
When file-offset is omitted, data is read from the beginning of the file.
This parameter is valid only when command qualifier /BINARY is specified.
Specifying other command qualifiers results in an error.

byte-count (data formula)

Specifies the loaded number of data by each bytes for the specified file name.
When byte-count is omitted, all data is read.
This parameter is valid only when command qualifier /BINARY is specified.
Specifying other command qualifiers results in an error.

● Command qualifiers

Specifies the file format, the operation when loading it, and the data length.

- File format specification

/OBJECT (default when omitted)

Loads load module file.

The default extension is ".abs".

/DEBUG

Loads only debug information from load module file.

The default extension is ".abs".

/BINARY

Loads binary format memory image file.

The default extension is ".bin".

Addressing cannot be omitted.

/COVERAGE

Loads coverage data file.

The default extension is ".cov".

This function is available only in the emulator debugger (MB2141).

/ALIAS

Loads alias file (command alias definition, macro definition).

The default extension is ".lst".

- Map setting specification

/AUTOMATIC (default when omitted)

For simulator debugger, automatically sets map area at loading.

/MANUAL

For simulator debugger, does not automatically set map area at loading.

A map area must be set by the SET MAP command previously.

/READ

For simulator debugger, sets ROM area for data segment as /READ attribute if AUTOMATIC qualifier is valid.

If this qualifier is omitted, the /READ/CODE attribute will be set.

- Fast loading specification (monitor debugger)

/FAST

This reduces the time to load the target file.

However, the flash memory area other than the target area for loading will be cleared because all the flash memory area must be cleared before loading.

/NOFAST (default when omitted)

Load the target file without reducing the loading time.

The flash memory area of the target area for loading will be cleared.

■ Description

The LOAD command loads the specified file.

This command can load the following four kinds of files. The file is opened adding the default extension respectively when the file name extension is omitted.

- Load module file

  Absolute-format object file created by linker.

- Memory image file

  Memory image file saved by SAVE command.
  An address should be always specified to load the files.

- Coverage data file

  Coverage data file saved by SAVE command.

- Alias file

  File containing command alias and macro definition.
  If a file name extension is omitted, the default extension is added and the file is opened.

■ Example

Other than monitor debugger

```
>LOAD debug
>LOAD/BINARY data.bin, FE00
```

Monitor debugger

```
>LOAD/FAST debug
```

Note:

When the fast loading is enabled in monitor debugger, all data loaded before will be cleared in order to clear all data in the flash memory area.

Therefore, when multiple modules are used in a project, put those modules in a single load module.

7.2 SAVE

When all command qualifiers are omitted or when /BINARY is specified, the SAVE command saves data in the specified memory to the memory image file (binary format of data only).

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SAVE file-name [, {address-range | module-name}]

● Parameters

file-name

Specify a name of file where memory data to be saved.

When the file name extension is omitted, any of the following extensions is added:

- ".bin" (valid when memory data saved in memory image)
- ".cov" (valid when coverage data saved)
- ".csv" (valid when the coverage measurement result saved in CSV format)
- ".lst" (valid when command alias or macro definition saved)

address-range (address formula)

Specify a memory area to be saved.

Address-range is valid only when command qualifier /BINARY is specified.

Specifying other command qualifiers results in an error.

module-name

Specifies a module name of the coverage measurement result to be saved.

This is valid only when the command qualifier is /COVERAGE.

If this parameter is omitted, the coverage rate of entire module is saved.

● Command qualifiers

- Kind of files

/BINARY (default when omitted)

Saves memory data to memory image file in binary format.

The default extension is ".bin".

Address-range specification cannot be omitted.

/COVERAGE

Saves coverage data in all areas specified by SET COVERAGE command.

The default extension is ".cov".

Address-range specification is invalid.

This function is available only in the emulator debugger (MB2141).

/ALIAS

Saves command alias definition and macro command definition to alias file.

The default extension is ".lst".

Address-range specification is invalid.

- Saving format

/CSV

The file is saved in CSV format.

This is valid only when the command qualifier is /COVERAGE.

■ Description

When all command qualifiers are omitted or when /BINARY is specified, the SAVE command saves data in the specified memory to the memory image file (binary format of data only).

In this case, address-range specification cannot be omitted.

If /COVERAGE is specified, this command will save coverage measurement data in all areas specified by the SET COVERAGE command.

If "/CSV" is specified at the same time, the coverage measurement result of the module is saved in CSV format.

When /ALIAS is specified, the command saves command alias definition and macro command definition to the alias file.

■ Example

```
>SAVE memo.bin, 0..0fff
>SAVE /COVERAGE/CSV   cov.csv, sample.c
```

CHAPTER 8

Source File/Symbol Commands

This chapter describes the Source File/Symbol commands.

8.1 LIST
8.2 SET PATH
8.3 SHOW PATH
8.4 SHOW SCOPE
8.5 UP
8.6 DOWN

8.1 LIST

The LIST command displays the source line corresponding to the specified line number.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

LIST [{ [file-name] line-number [.. line-number] | address}]

● Parameters

file-name

Specify a name of source file to be displayed.

When file-name is omitted, the previously-specified file name is assumed.

line-number

Specify a number of source line to be displayed.

"$" must always precede a line number.

When line numbers are delimited by "..", the source lines within the specified range are displayed.

address (address formula)

Specify an address (memory location) where code attribute is stored.

Specify this parameter when displaying the source line corresponding to the address (memory location).

■ Description

The LIST command displays the source line corresponding to the specified line number.

When only a period (.) is specified in parameter, the command displays starting from the source line corresponding to the current program counter.

If the value in the program counter is rewritten due to program execution when all parameters are omitted, the command starts source line display from the source line corresponding to the current program counter.

In other cases, the command displays 19 source lines, starting from the line next to the previously-displayed last line.

■ Example

```
>LIST PROGRAM.C$2..$3
2:    x = x+1 ;
3:    printf ("%d\n", x) ;
>LIST subdisp
30:   subdisp ( )
31:   {
32:   int i;
33:
34:         for (i = p; i  > = 1; i--)
35:         printf ( "data [%d] = %d \n", i, data [i] );
36:
     .
     .
     .
>LIST.
53:   switch (*s) {
54:    case '0' : z = " "; return (z) ;
55:    case '1' : z = "a"; return (z) ;
56:    case '2' : z = "b"; return (z) ;
     .
     .
```

8.2 SET PATH

The SET PATH command specifies the directories used to search for the source file.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SET PATH [source-search-directory-name [, ...]]

● Parameter

source-search-directory-name

Specify a directory for which source file is searched.

● Command qualifier

/APPEND

Appends specified search directory to current setting.

■ Description

The SET PATH command specifies the directories used to search for the source file.

The command searches the specified directories for the source file in sequence from the left.

When user omit a parameter, a debugger strikes registered source-search-directory-name off (A search directory becomes a current directory).

■ Example

```
>SET PATH A:\
```

8.3 SHOW PATH

The SHOW PATH command displays currently-enabled source file search directories.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW PATH

■ Description

The SHOW PATH command displays currently-enabled source file search directories.

■ Example

```
>SHOW PATH
source file search path = a:\
```

8.4 SHOW SCOPE

The SHOW SCOPE command displays the module and function names including the memory location indicated by the current program counter.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SHOW SCOPE

■ Description

The SHOW SCOPE command displays the module and function names including the memory location indicated by the current program counter.

■ Example

```
>SHOW SCOPE
current scope = SIEVE\sub_main\
```

8.5 UP

The UP command moves the scope to the parent function.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

UP

■ Description

The UP command moves the scope to the parent function.

UP/DOWN information is cleared when the MCU is executed, RESET is performed, or the program counter is updated.

■ Example

```
>UP
Current Scope = demo\sort\
```

8.6 DOWN

The DOWN command moves the scope to the child function.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

DOWN

■ Description

The DOWN command moves the scope to the child function.

UP/DOWN information is cleared when the MCU is executed, RESET is performed, or the program counter is updated.

■ Example

```
>DOWN
Current Scope = demo\check\
```

CHAPTER 9

Command Procedure Commands

This chapter describes the Command Procedure commands.

9.1 BATCH
9.2 QUIT

9.1 BATCH

The BATCH command executes the commands in the specified command procedure file.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

BATCH file-name [, actual-parameter [, ...]]

● Parameters

file-name

Specify a name of file where command procedure to be executed is written.

The default extension is ".prc".

actual-parameter

Specify an actual parameter required for command procedure.

● Command qualifier

/ICON

Converts debugger to icon and executes it when command procedure executed.

When command procedure execution terminates, the icon is restored to the original size.

■ Description

The BATCH command executes the commands in the specified command procedure file.

Batch processing (procedure file call) can be nested for up to 8 levels.

Actual parameters are replaced with temporary parameters in the command procedure file (%P0 to %P9) in the order they were specified.

When the count of temporary parameters is greater than that of the specified actual parameters, the remaining temporary parameters are replaced by empty character strings, null.

When the count of temporary parameters is less than that of the specified actual parameters, the remaining parameters are ignored.

The count of the specified actual parameters can be referred by means of %NP.

■ Example

```
>BATCH TST.PRC, 0, 0FFF, BRK
```

9.2 QUIT

Executing the QUIT command when the command procedure is being executed quits command procedure processing.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

QUIT

■ Description

Executing the QUIT command when the command procedure is being executed quits command procedure processing.

■ Example

Data in command procedure file

```
IF %NP  < 2
     QUIT
ENDIF
SET VARIABLE I = 0
SET VARIABLE ADDR = %P0
WHILE %I <%P1
     SET MEMORY %ADDR = %I
     SET VARIABLE I = %I+1
     IF %ADDR == H'FFFF
        QUIT
     ELSE
        SET VARIABLE ADDR = %ADDR+1
     ENDIF
ENDW
```

CHAPTER 10

Replacement Commands

This chapter describes the Replacement commands.

10.1 SET ALIAS
10.2 SHOW ALIAS
10.3 CANCEL ALIAS
10.4 SET VARIABLE
10.5 SHOW VARIABLE
10.6 CANCEL VARIABLE

10.1 SET ALIAS

The SET ALIAS command defines a command alias.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SET ALIAS alias = command-character-string

● Parameters

alias (identifier)

Specify a command alias.

command-character-string

Specify a command character string (command name, command qualifier, and parameter) to be replaced with specified alias, enclosed in double quotation marks (").

■ Description

The SET ALIAS command defines a command alias.

It is convenient to define command aliases for frequently-used commands.

No command alias can be nested.

Other command aliases cannot be included in command alias definition.

■ Example

```
>SET ALIAS BP = "SET BREAK FF00,3"
>SET ALIAS E = "ENTER"
>SET ALIAS R = "SHOW REGISTER"
```

10.2 SHOW ALIAS

The SHOW ALIAS command displays the defined command alias list.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW ALIAS

■ Description

The SHOW ALIAS command displays the defined command alias list.

■ Example

```
>SHOW ALIAS
T    :    STEP
D    :    EXAMINE
PC   :    SHOW REGISTER PC
>
```

10.3 CANCEL ALIAS

The CANCEL ALIAS command cancels the alias of the specified command character string.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

CANCEL ALIAS [alias [, ...]]

● Parameter

alias (identifier)

Specify a command alias to be cancelled.

● Command qualifier

/ALL

Cancels aliases of all command character strings.

■ Description

The CANCEL ALIAS command cancels the alias of the specified command character string.

■ Example

```
>CANCEL ALIAS BP
>
```

10.4 SET VARIABLE

The SET VARIABLE command defines a debug variable.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SET VARIABLE debug-variable-name = replacing-character-string

● Parameters

debug-variable-name (identifier)

Specify a debug variable to be defined.

replacing-character-string

Specify a character string replacing debug variable.

■ Description

The SET VARIABLE command defines a debug variable.

The defined debug variable can be used as part of the parameter field when the command is specified.

The used debug variable is replaced with the replacing character string defined by this command as it is.

All the variables that can be specified in the parameter field can be defined.

For example, a character string and an expression can be defined as they are.

The debug variable is replaced with the defined character string as it is, so take care when defining with the same debug variables (e.g., increment).

■ Example

```
>SET VARIABLE ADDR = 0309+12
>SET VARIABLE STR = "ABCDEF"
>SET MEMORY/STRING %ADDR = %STR
```

(can be replaced with SET MEMORY/STRING 0309+12 = "ABCDE")

```
>SET VARIABLE CNT = 1
>WHILE %CNT <5
*PRINTF "val [%d] = %d\n", %CNT, %CNT
*SET VARIABLE CNT = %EVAL (%CNT+1)
```

(The %EVAL function is defined so that the CNT character string will not exceed the limit.)

```
*ENDW
```

10.5 SHOW VARIABLE

The SHOW VARIABLE command displays the definition of the specified debug variable.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW VARIABLE [debug-variable-name [, ...]]

● Parameter

debug-variable-name (identifier)

Specify a debug variable name to be displayed.

● Command qualifier

/ALL (default when omitted)

Displays all debug variables.

■ Description

The SHOW VARIABLE command displays the definition of the specified debug variable.

■ Example

```
>SET VARIABLE CNT = 1
>WHILE %CNT <5
*SHOW VARIABLE CNT
*PRINTF "CNT = %d\n", %CNT
*SET VARIABLE CNT = %CNT+1
*ENDW
CNT : 1
CNT = 1
CNT : 1+1
CNT = 2
CNT : 1+1+1
CNT = 3
CNT : 1+1+1+1
CNT = 4
```

10.6 CANCEL VARIABLE

The CANCEL VARIABLE command cancels the specified debug variable.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

CANCEL VARIABLE [debug-variable-name [, ...]]

● Parameter

debug-variable-name (identifier)

Specify a debug variable name to be cancelled.

● Command qualifier

/ALL

Cancels all debug variables.

■ Description

The CANCEL VARIABLE command cancels the specified debug variable.

■ Example

```
>CANCEL VARIABLE CHKADR, X, Y
```

CHAPTER 11

Utility Commands

This chapter describes the Utility commands.

11.1 SET LOGGING
11.2 SHOW LOGGING
11.3 CANCEL LOGGING
11.4 ENABLE LOGGING
11.5 DISABLE LOGGING
11.6 PRINTF
11.7 SET OUTPUT
11.8 SHOW OUTPUT

11.1 SET LOGGING

The SET LOGGING command opens the specified logging file and starts logging.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SET LOGGING [file-name]

● Parameter

file-name

Specify a log file name.
The default extension is ".LOG".
When file-name is omitted, the DEBUG.LOG file is used to log data.

● Command qualifiers

/OPEN (default when omitted)

Newly opens specified file.

/APPEND

Appends log data to end of specified file.

/EXPANSION (default when omitted)

Logs command list and its result.

/UNEXPANSION

Logs only result.

/COMMAND

Logs only user-entered data.

■ Description

The SET LOGGING command opens the specified logging file and starts logging.

When command qualifier /APPEND is specified, data in the previous file is not lost.

The data to be logged can be selected.

When command qualifier /COMMAND is specified, the entered command is only logged. The output file is used as the command procedure file.

■ Example

```
>SET LOGGING filename.log
>
>SET LOGGING/COMMAND filename.log
```

11.2 SHOW LOGGING

The SHOW LOGGING command displays the logging status.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW LOGGING

■ Description

The SHOW LOGGING command displays the logging status.

■ Example

```
>SHOW LOGGING
en/dis         :     ENABLE
logging file   :     logfile.log
logging data   :     EXPANSION
```

11.3 CANCEL LOGGING

The CANCEL LOGGING command cancels the logging setup and closes the logging file.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

CANCEL LOGGING

■ Description

The CANCEL LOGGING command cancels the logging setup and closes the logging file.

■ Example

```
>CANCEL LOGGING
```

11.4 ENABLE LOGGING

The ENABLE LOGGING command enables logging.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

ENABLE LOGGING

■ Description

The ENABLE LOGGING command enables logging.

■ Example

```
>ENABLE LOGGING
```

11.5 DISABLE LOGGING

The DISABLE LOGGING command temporarily disables logging.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

DISABLE LOGGING

■ Description

The DISABLE LOGGING command temporarily disables logging.

The ENABLE LOGGING command can be used to enable logging again.

■ Example

```
>DISABLE LOGGING
```

11.6 PRINTF

The PRINTF command displays the specified character string and the expression value of the specified format on the screen.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

PRINTF "format-control-string" [, expression [, ...]]

● Parameters

format-control-string

Specify character strings to be displayed on screen and format for expression value display.

Enclose format specification in double quotation marks (").

"% [flag] [width] [.precision] [l] type"

%

Specify this parameter when displaying data according to format specification.

The PRINTF command displays characters that are not format specification after % as they are.

flag

Specify whether to right- or left-justify display, o (octal number) or 0x, 0X (hexadecimal number), and output control.

When flag is omitted, the display is right-justified.

This parameter is invalid when the conversion display format type is b or f.

- : Left-justification

: Adds 0, 0x, or 0X before numeric value. 0 is added when the conversion display format is o. 0x is added when the format is x. 0X is added when the format is X.

width

Specify minimum count of digits of integral to be output.

When the conversion result is less than the specified count of digits, the remaining areas are padded with space.

To pad with 0s at right-justification, add 0 to the beginning and specify the digits count.

When the conversion display format type is b or f, width is invalid.

precision

Specify minimum count of digits of integer to be output.

When the conversion result is less than the specified count of digits, the remaining areas are padded with 0s.

When the conversion display format type is b or f, precision is invalid.

l

Specify whether to display the language expression value as the long, unsigned long type when the conversion display format type is d, u, o, x, X.

When l is not specified, the language expression value is assumed to be the short, unsigned short type.

type

Specify one of following conversion display formats for the value of expressions:

d : Signed decimal number

u : Unsigned decimal number

o : Unsigned octal number

x : Unsigned hexadecimal number (Lower-case characters a to f represent 10 to 15, respectively.)

X : Unsigned hexadecimal number (Upper-case characters A to F represent 10 to 15, respectively.)

c : One character

b : Unsigned binary number

s : Character string (Only addressing is valid. The maximum number of characters is 128 bytes.)

expression

Specify the expression to be displayed.

■ Description

The PRINTF command displays the specified character string and the expression value of the specified format on the screen.

■ Example

```
>PRINTF "ABC = %d\n", datflg
ABC = 3
```

11.7 SET OUTPUT

When the user program stops, the SET OUTPUT command opens the source window according to the debug information at the position indicated by the PC.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SET OUTPUT

● Command qualifiers

/SOURCE (default when omitted)

Opens source window in mixed mode, even if no file.

/INSTRUCTION

Opens source window as disassembly window, even if no file.

■ Description

When the user program stops, the SET OUTPUT command opens the source window according to the debug information at the position indicated by the PC. In this case, the operation that is performed when no target source file can be found is set.

■ Example

```
>SET OUTPUT /SOURCE
```

11.8 SHOW OUTPUT

The SHOW OUTPUT command shows the display mode set by the SET OUTPUT command.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

SHOW OUTPUT

■ Description

The SHOW OUTPUT command shows the display mode set by the SET OUTPUT command.

■ Example

```
>SHOW OUTPUT
source mode:  source
```

CHAPTER 12

Task Debug Commands

This chapter describes the Task Debug commands.

12.1 SHOW OBJECT

12.1 SHOW OBJECT

The SHOW OBJECT command displays specified object data.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

SHOW OBJECT [object-number]

● Parameter

object-number (valid only for /TSK, /SEM, /FLG, /MBX, /MPL, /CYC, and /ALM)

Specify an object number to be displayed.

When object-number is omitted, the outlines of all the objects are displayed.

● Command qualifiers

- Object specification

/TSK

Displays task data.

/SEM

Displays semaphore data.

/FLG

Displays an event flag data.

/MBX

Displays mailbox data.

/CYC

Displays cyclic handler data.

/RDYQ

Displays ready queue data.

/TMRQ

Displays data in timer wait queue.

■ Description

The SHOW OBJECT command displays specified object data.

■ Example

```
>SHOW OBJECT/TSK 1
< Task Information >
[tskid]   0001    [tcbadr]   2100    [tskpri]   02   [itskpri]   02
[status]  READY
[wupcnt]  00      [tmocnt]   0000
[stack]   2347    [stkarea]  2253..2352

>SHOW OBJECT/SEM 1
< Semaphore Information >
[semid]   0001    [QUEadr]   2191
[semcnt]  0001    [semadr]   2189
[wtskid]  NONE

>SHOW OBJECT/FLG 1
< EventFlag Information >
[flgid]   0001    [QUEadr]   2169
[flgptn]  clear   [flgadr]   2168:0
[wtskid]  NONE

>SHOW OBJECT/MBX 1
< Mailbox Information >
[mbxid]   0001    [QUEadr]   21b1
[msgadr]  NONE
[wtskid]  NONE

>SHOW OBJECT/CYC 1
< Cyclic Handler Information >
[cycid]   0001    [cycadr]   21d1
[cycact]  -----   [cychdr]   -----
[1fttim]  0000    [cyctime]  0000

>SHOW OBJECT/RDYQ
< Priority Ready Queue Information >
[prino]   01      [tskcnt]   0000
[prino]   02      [tskcnt]   0000
[prino]   03      [tskcnt]   0001
[prino]   04      [tskcnt]   0001
```

```
>SHOW OBJECT/TMRQ
< Timer Queue Information >
[tmrqadr]  2249
[id]       T0001  C0014
```

CHAPTER 13

Control Commands

This chapter describes the Control commands.

13.1 IF
13.2 REPEAT
13.3 WHILE
13.4 BREAK

13.1 IF

When formula is evaluated as true, the command list immediately after IF is executed. When formula is evaluated as false, the command list after ELSE is executed.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

```
IF formula
  command-list
[ELSEIF formula
  command-list]
[ELSE
  command-list]
ENDIF
```

● Parameters

formula

Specify the execution condition formula of specified command list.

command-list

Specify the commands to be executed.

■ Description

When formula is evaluated as true, the command list immediately after IF is executed. When formula is evaluated as false, the command list after ELSE is executed.

If formula is false when ELSE is omitted, nothing is executed.

Only macros or batch can use the IF command.

■ Example

```
IF %R0 == 0
  print "OK!!"
else
  print "NG!!"
endif
```

13.2 REPEAT

The REPEAT command evaluates the UNTIL formula after the command list specified by command-list has been executed. This command repeats execution of the command list while the formula is false.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

REPEAT

command-list

UNTIL formula

● Parameters

command-list

Specify the commands to be executed.

formula

Specify the execution condition formula of specified command list.

■ Description

The REPEAT command evaluates the UNTIL formula after the command list specified by command-list has been executed. This command repeats execution of the command list while the formula is false.

Only macros or batch can use the REPEAT command.

■ Example

```
REPEAT
     STEP
UNTIL  %PC == main
```

13.3 WHILE

When the specified formula is evaluated as true, the WHILE command repeats execution of the specified command list.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

```
WHILE formula
  command-list
ENDW
```

● Parameters

formula

Specify the execution condition formula of specified command list.

command-list

Specify the commands to be executed.

■ Description

When the specified formula is evaluated as true, the WHILE command repeats execution of the specified command list.

Only macros or batch can use the WHILE command.

■ Example

```
WHILE %PC! = function
     STEP
ENDW
```

13.4 BREAK

The BREAK command enables the program to exit the control structure.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

BREAK

■ Description

The BREAK command enables the program to exit the control structure.

This command is valid only in the REPEAT and WHILE command lists.

Only macros or batch can use the BREAK command.

■ Example

```
WHILE    1
  if %PC == main
  BREAK
  ENDIF
  STEP
ENDW
```

CHAPTER 14

Built-in Variables and Functions

This chapter describes the Built-in Variables and Functions.

14.1 %CALL

14.2 %ERRNUM

14.3 %ENTRY

14.4 %STKTOP

14.5 %RADIX

14.6 %SCPADR

14.7 %LOADNUM

14.8 %BIT, %B, %W, %L, %S, %D

14.9 %STRGET

14.10 %STRSTR

14.11 %STRCMP

14.12 %STRLEN

14.13 %STRCAT

14.14 %SYMLEN

14.15 %TOVAL

14.16 %TOSTR

14.17 %EVAL

14.1 %CALL

%CALL replaces the return value with the last-executed CALL command.

■ Debugger

Simulator		○
Emulator	(MB2141)	○
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

%CALL

■ Description

%CALL replaces the return value with the last-executed CALL command. If the function return values are void and double, 0 is returned.

■ Example

```
>CALL func(100,200)
return value is H'40
>ENTER val=%CALL+0x80
```

14.2 %ERRNUM

%ERRNUM replaces the error number with the last error number executed from the Command Window.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%ERRNUM

■ Description

%ERRNUM replaces the error number with the last error number executed from the Command Window.

0 indicates that there is no error.

■ Example

```
>PRINTF "ERROR NO. = %d\n", %ERRNUM
ERROR NO. = 5
```

14.3 %ENTRY

%ENTRY replaces the execution starting address of the loaded module.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%ENTRY

■ Description

%ENTRY replaces the execution starting address of the loaded module.

0 indicates that there is no execution starting entry.

■ Example

```
>PRINTF "ENTRY = 0x%X\n", %ENTRY
ENTRY = 0x1000
```

14.4 %STKTOP

%STKTOP replaces the starting address of the stack area for the loaded module.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%STKTOP

■ Description

%STKTOP replaces the starting address of the stack area.

0 indicates that there is no stack area.

■ Example

```
>PRINTF "STACK = 0x%X\n", %STKTOP
STACK = 0x8000
```

14.5 %RADIX

%RADIX replaces the base number with the currently-set base number (BINARY, OCTAL, DECIMAL, or HEXADECIMAL).

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%RADIX

■ Description

%RADIX replaces the base number with the currently-set base number (BINARY, OCTAL, DECIMAL, or HEXADECIMAL).

■ Example

```
>PRINTF "base-number = "
>PRINTF %TOSTR(%RADIX)
base-number = HEXADECIMAL
```

14.6 %SCPADR

%SCPADR replaces the current scope address.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%SCPADR

■ Description

%SCPADR replaces the current scope address.

■ Example

```
>PRINTF " scope = 0x%X\n", %SCPADR
scope = 0x1830
```

14.7 %LOADNUM

%LOADNUM replaces the size of the last loaded binary file.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%LOADNUM

■ Description

%LOADNUM replaces the size of the last loaded binary file.

■ Example

```
>PRINTF "byte-count = %d\n", %LOADNUM
byte-count = 584
```

14.8 %BIT, %B, %W, %L, %S, %D

%BIT, %B, %W, %L, %S, or %D replaces the data with any of the following memory data read from the specified address.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	○
	(MB2146-08)	○
	(MB2146-07)	○
Monitor		○

■ Format

%BIT(address)

%B(address)

%W(address)

%L(address)

%S(address)

%D(address)

● Parameter

address

Specify the address from where memory data to be read.

■ Description

%BIT, %B, %W, %L, %S, or %D replaces the data with any of the following memory data read from the specified address:

%BIT: Bit data
%B: Byte (8bit) data
%W: Word (16bit) data
%L: Long word (32bit) data
%S: Single-precision floating-point number data
%D: Double-precision floating-point number data

■ Example

```
>PRINTF "1000 = 0x%X\n", %W(1000)
1000 = 0x3020
```

14.9 %STRGET

%STRGET replaces the character string in the specified count of characters, starting from the specified character position in the specified character string.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%STRGET(character-string, character-position, character-count)

● Parameters

character-string

Specify a character string to be replaced.

character-position

Specify a character position where get processing to be replaced (character position relative to first character).

character-count

Specify a count of characters to be replaced.

■ Description

%STRGET replaces the character string in the specified count of characters, starting from the specified character position in the specified character string.

■ Example

```
>PRINTF %TOSTR(%STRGET("abcdefghijklmn", 3, 4) )
cdef
```

14.10 %STRSTR

%STRSTR checks whether character-string-1 includes character-string-2.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%STRSTR(character-string-1, character-string-2)

● Parameters

character-string-1

Specify a character string to be candidated.

character-string-2

Specify a character string to be searched.

■ Description

%STRSTR checks whether character-string-1 includes character-string-2.

When character-string-1 includes character-string-2, %STRSTR replaces with the character position number in character-string-1.

When character-string-1 does not include character-string-2, %STRSTR replaces the character position number with 0.

■ Example

```
>PRINTF "%d\n", %STRSTR("abcdefghijklmn", "fg")
6
```

14.11 %STRCMP

%STRCMP compares character-string-1 with character-string-2.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%STRCMP(character-string-1, character-string-2)

● Parameter

character-string-1,character-string-2

Specify a character strings to be compared.

■ Description

%STRCMP compares character-string-1 with character-string-2.

When the character-string-1 matches with character-string-2, %STRCMP sets to 0. When the character-string-1 does not matches with character-string-2, %STRCMP sets to 1.

■ Example

```
>PRINTF "%d\n", %STRCMP("abcde", "fg")
1
>PRINTF "%d\n", %STRCMP("abcde", "abcde")
0
```

14.12 %STRLEN

%STRLEN replaces character string with the count of characters.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%STRLEN(character-string)

● Parameter

character-string

Specify a character string to be replaced.

■ Description

%STRLEN replaces character string with the count of characters.

■ Example

```
>PRINTF "%d\n", %STRLEN("abcde")
5
```

14.13 %STRCAT

%STRCAT replaces character string created by linking character-string-1 and character-string-2.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%STRCAT(character-string-1, character-string-2)

● Parameter

character-string-1, character-string-2

Specify a character strings to be linked.

■ Description

%STRCAT replaces character string created by linking character-string-1 and character-string-2.

■ Example

```
>PRINTF %TOSTR(%STRCAT ("abcde", "fg") )
abcdefg
```

14.14 %SYMLEN

%SYMLEN returns the size of a specified symbol.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%SYMLEN(symbol-name)

● Parameter

symbol-name

Specify a symbol.

■ Description

%SYMLEN returns the size of a specified symbol.

■ Example

```
>PRINTF "%d\n", %SYMLEN("abcde")
2
```

14.15 %TOVAL

%TOVAL deletes double quotation marks (") from both ends of the specified character string.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%TOVAL(character-string)

● Parameter

character-string

Specify a character string.

■ Description

%TOVAL deletes double quotation marks (") from both ends of the specified character string.

This function is used when the character string enclosed in double quotation marks is specified in a field where only parameters other than character strings can be written.

■ Example

```
>SET BREAK %TOVAL("main")
```

14.16 %TOSTR

%TOSTR encloses the specified character string in double quotation marks (").

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%TOSTR(character-string)

● Parameter

character-string

All parameter types can be specified.

■ Description

%TOSTR encloses the specified character string in double quotation marks (").

This function is used when the specified character string is specified in a field where only character strings can be written as parameters.

■ Example

```
>PRINTF %TOSTR(main)
main
```

14.17 %EVAL

%EVAL evaluates specified expression.

■ Debugger

Simulator		◎
Emulator	(MB2141)	◎
	(MB2146-09/09A/09B)	◎
	(MB2146-08)	◎
	(MB2146-07)	◎
Monitor		◎

■ Format

%EVAL(expression)

● Parameter

expression

Specify an expression to be evaluated.

■ Description

%EVAL evaluates specified expression.

■ Example

```
>PRINTF "%d\n", %EVAL(10+20+30)
60
```

APPENDIX

These appendixes describe the Manager-Related Messages, Error Message for Debuggers, and Execution Suspension Messages List.

APPENDIX A Manager-Related Messages
APPENDIX B Error Message for Debuggers
APPENDIX C Execution Suspension Messages List

APPENDIX A Manager-Related Messages

This appendix describes Manager-Related.

- Manager-Related Messages

E4002W	Insufficient memory.

"Explanation" System memory is insufficient.
"Operator response" Terminate another program and execute this program.

E4011W	Registration not possible.

"Explanation" Data cannot be written to the system registry.
"Operator response" Terminate another program and execute this program.

E4012W	Failed function call. Exe file is old.

"Explanation" The version of the program file does not correspond to that of the DLL file.
"Operator response" Install the latest version of SOFTUNE Workbench.

E4013W	Failed function call. DLL file is old.

"Explanation" The version of the program file does not correspond to that of the DLL file.
"Operator response" Install the latest version of SOFTUNE Workbench.

E4020W	CPU information file version is different. Contains uninterpretable information.

"Explanation" The CPU information file is old and does not contain the required information.
"Operator response" Get the latest CPU information file.

E4021W	Chip type in CPU information file is not applicable.

"Explanation" Information file for a different CPU is specified.
"Operator response" Specify the correct CPU information file.

E4022W	Please enter CPU information file.

"Explanation" The CPU information file cannot be found.
"Operator response" Enter the CPU information file directory.

E4023W	Illegal tool option data. Default data is set.

"Explanation" The project file has illegal value written tool option data.
"Operator response" Reset the tool option data.

E4024W	Invalid CPU information. Set default value.

"Explanation" The CPU information file has illegal data.
"Operator response" Get the latest CPU information file.

E4100W	Access was denied.

"Explanation" The file cannot be accessed.
"Operator response" The file may be write- or read-disabled. Check the file attributes.

E4110W	Too many open files.

"Explanation" The maximum number of files that can be opened is exceeded.
"Operator response" Close other files.

E4120W	Directory does not exist.

"Explanation" The directory cannot be found.
"Operator response" Enter the correct directory name.

E4121W	Drive is not ready.

"Explanation" The drive cannot be accessed.
"Operator response" Check the drive.

E4122W	Path is invalid.

"Explanation" The directory cannot be found.
"Operator response" Enter the correct directory name.

E4123W	Unable to create directory.

"Explanation" The directory cannot be created.
"Operator response" The directory may be write-disabled, or directory name may be incorrect.

E4124W	Unable to delete directory.

"Explanation" The directory cannot be deleted.
"Operator response" The directory may be write-disabled, or a file in the directory may be in use by another process.

E4125W	Destination disk is full.

"Explanation" The remaining capacity of the disk is insufficient.
"Operator response" Delete unnecessary files.

E4126W	Could not be removed because it is the current directory.

"Explanation" An attempt was made to delete the current directory.
"Operator response" Move from the current directory to delete another directory.

E4127W	This directory cannot be access.

"Explanation" Access to the directory is denied.
"Operator response" Permission to access the directory may be denied.

E4130W	File cannot be open.

"Explanation" The file cannot be opened.
"Operator response" Permission to access the file or directory may be denied.

E4131W	File cannot be close.

"Explanation" The file cannot be closed.
"Operator response" Permission to access the file or directory may be denied.

E4132W	File cannot be read.

"Explanation" The file cannot be read.
"Operator response" Permission to access the file or directory may be denied.

E4133W	File cannot be written.

"Explanation" The file cannot be written.
"Operator response" Permission to access the file or directory may be denied.

E4134W	File cannot be create.

"Explanation" The file cannot be created.
"Operator response" Permission to access the file or directory may be denied.

E4135W	File cannot be delete.

"Explanation" The file cannot be deleted.
"Operator response" Permission to access the file or directory may be denied.

E4136W	File cannot be change name.

"Explanation" The file cannot be renamed.
"Operator response" Permission to access the file or directory may be denied.

E4137W	File cannot be copied.

"Explanation" The file cannot be copied.
"Operator response" Permission to access the file or directory may be denied.

E4138W	File not found.

"Explanation" The file cannot be found.
"Operator response" Check the file name.

E4140W	File not found. Do you create this file?

"Explanation" The file cannot be found.
"Operator response" To create a new file, click the OK button.

E4142W	A sharing violation occurred while accessing.

"Explanation" The same file is being used by another process.
"Operator response" Terminate the other program. In some rare cases, the file may remain in use even after the program is terminated. In this case, reboot Windows.

E4143W	A locking violation occurred while accessing.

"Explanation" The same file is being used by another process.
"Operator response" Terminate the other program. In some rare cases, the file may remain in use even after the program is terminated. In this case, reboot Windows.

E4200W	The project file format is illegal.

"Explanation" The project file cannot be read properly.
"Operator response" The project file may be different from that for SOFTUNE Workbench or may be damaged. Create a new project file.

E4201W	Project file cannot be opened - CPU type is different.

"Explanation" The project file is different family for the MCU.
"Operator response" Create a new project file for the MCU.

E4202W	Unable to save project file.

"Explanation" An error occurred at writing to the project file.
"Operator response" The remaining disk capacity may be insufficient or the project file may be write-disabled.

E4204W	Illegal CPU information of project file. Setting default value.

"Explanation" CPU information in the project file is illegal, and is substituted for the default.
"Operator response" Check the set value for CPU information in the project file.

E4205W	Target file directory not found. Create a directory?

"Explanation" The target project file writing directory is not specified.
"Operator response" Click the OK button to create a directory.

E4206W	List file directory not found. Create a directory?

"Explanation" The target list file writing directory is not specified.
"Operator response" Click the OK button to create a directory.

E4207W	Object file directory not found. Create a directory?

"Explanation" The target object file writing directory is not specified.
"Operator response" Click the OK button to create a directory.

E4210W	Please specify the project name.

"Explanation" The project name is not specified.
"Operator response" Enter the project name.

E4211W	Please specify the project directory.

"Explanation" The project directory is not specified.
"Operator response" Enter the project directory name.

E4212W	Please specify the target file name.

"Explanation" The target file name is not specified.
"Operator response" Enter the target file name.

E4213W	Includes characters that cannot be designated. \ / : , ; * ? " " < > |

"Explanation" These characters cannot be used.
"Operator response" Change the name.

E4214W	Includes characters that cannot be designated. , ; * ? " " < > |

"Explanation" These characters cannot be used.
"Operator response" Change the name.

E4215W	Includes characters that cannot be designated. , ; * ? " " < > |

"Explanation" These characters cannot be used.
"Operator response" Change the name.

E4220W	Please specify the target file name.

"Explanation" The target file name is not specified.
"Operator response" Enter the target file name.

E4221W	Directory not found. Do you create this directory?

"Explanation" The directory is not found.
"Operator response" Enter the directory name.

E4222W	Unable to create directory.

"Explanation" The directory cannot be created.
"Operator response" The file may be write-disabled.

E4223W	Changed target MCU. CPU information changed to default value.

"Explanation" When the target MCU is changed, the preset CPU information returns to the default value.
"Operator response" Reset the CPU information.

E4224W	Specify target MCU.

"Explanation" The target MCU is not specified.
"Operator response" Enter the target MCU name.

E4225W	Specify project type.

"Explanation" The project type is not specified.
"Operator response" Specify the project type.

E4226W	Includes characters that cannot be designated. , ; * ? " " < > |

"Explanation" These characters cannot be used.
"Operator response" Change the name.

E4227W	Please specify Object File Directory.

"Explanation" The target object file writing directory is not specified.
"Operator response" Enter the directory name.

E4228W	Please specify List File Directory.

"Explanation" The target list file writing directory is not specified.
"Operator response" Enter the directory name.

E4230W	Double specification.

"Explanation" The same specification is already in use.
"Operator response" Change the specification.

E4232W	Setup file is not registered. Registered automatically.

"Explanation" Starting the debugger requires a setup file. If a setup file is not specified, create it with the same name as that of the project name.
"Operator response" Use [Project]-[Setup Project]-[Debug] to set the items required for the automatically-created setup file.

E4233W	Available setup file is not registered. Registered automatically.

"Explanation" Starting the debugger requires a setup file. If a setup file is not specified, create it with the same name as that of the project name.

"Operator response" Use [Project]-[Setup Project]-[Debug] to set the items required for the automatically-created setup file.

E4234W	Please specify the title.

"Explanation" The title is not specified.

"Operator response" Specify the title.

E4240W	Already a registered member.

"Explanation" The specified file is already saved in the project.

"Operator response" Check the file name.

E4241W	This file name has already been registered.

"Explanation" The specified file has been already registered in the project.

"Operator response" Check the file name.

E4242W	File not found. Do you registered?

"Explanation" An attempt was made to register a non-existent file in the project.

"Operator response" If the file name is correct, register the file. An inquiry is made as to whether to create a new file when starting the editor.

E4243W	Too many select files.

"Explanation" The count of selected files exceeds the maximum value.

"Operator response" Decrease the count of selected files.

E4301W	Unable to create command line.

"Explanation" The option file to start the language tool cannot be created.

"Operator response" Check the access permission for the OPT subdirectory under the project directory, or the disk capacity.

E4302W	Failed during start.

"Explanation" The tool cannot be started.
"Operator response" The tool name may be incorrect. Check the tool settings.

E4303W	Command Line too long.

"Explanation" The command line is too long (max. 2048 characters).
"Operator response" Check the option parameters.

E4304W	Failed during start editor.

"Explanation" The registered external editor cannot be started.
"Operator response" Check the executable file name of the editor.

E4305W	Compiler/Assembler is started.

"Explanation" An attempt is made to close the project during tool start up.
"Operator response" Use the Suspend button to terminate the tool and close the project.

E4306W	Make function is started.

"Explanation" An attempt is made to close the project during tool start up.
"Operator response" Use the Suspend button to terminate the tool and close the project.

E4307W	Build function is started.

"Explanation" An attempt is made to close the project during tool start up.
"Operator response" Use the Suspend button to terminate the tool and close the project.

E4308W	Include Dependencies is started.

"Explanation" An attempt is made to close the project during tool start up.
"Operator response" Use the Suspend button to terminate the tool and close the project.

E4309W	Tool is started.

"Explanation" An attempt is made to close the project during tool start up.
"Operator response" Use the Suspend button to terminate the tool and close the project.

E4400W	Setup file is read only. Setup information is not saved.

"Explanation" The setup file cannot be written.
"Operator response" Set the setup file to the write-enabled state.

E4401W	Setup file not found.

"Explanation" The setup file was not found.
"Operator response" It may not be valid that the Softune Workbench was installed completely. Install it again.

E4402W	The setup file format is illegal.

"Explanation" The setup file is written invalid data.
"Operator response" It may not be valid that the SOFTUNE Workbench was installed completely. Install it again.

E4420W	Maximum of address is xxxx.

"Explanation" The address exceeds the maximum value.
"Operator response" Check the address specification.

E4421W	The start address exceeds the end address.

"Explanation" The specified address range is incorrect.
"Operator response" Check the address range specification.

E4422W	The designated address is already designated.

"Explanation" The specified address range has been already registered.
"Operator response" Check the address range.

E4601W	Double specification.

"Explanation" The specified item has been already existent.
"Operator response" Check the specification contents.

E4603W	Illegal tool option data.

"Explanation" The tool option data does not have the necessary data.
"Operator response" Open the Tool Option Check dialog and click the OK button. When the control data is displayed, input the necessary data.

E4604W	There is no control data.

"Explanation" Unspecified control data is found.
"Operator response" Specify the control data.

E4605W	Includes characters that cannot be designated.

"Explanation" These characters cannot be used.
"Operator response" Change the name.

E4606W	Includes characters that cannot be designated. , ; * ? " " < > |

"Explanation" These characters cannot be used.
"Operator response" Change the name.

E4607W	Includes characters that cannot be designated. , ; * ? " " < > |

"Explanation" These characters cannot be used.
"Operator response" Change the name.

E4610W	The range of the number of lines is 20-255.

"Explanation" The count of lines exceeds the limit.
"Operator response" Change the count of lines.

E4611W	The range of the number of columns is 80-1023.

"Explanation" The count of columns exceeds the limit.
"Operator response" Change the count of columns.

E4612W	The range of the number of columns is 70-1023.

"Explanation" The count of columns exceeds the limit.
"Operator response" Change the count of columns.

E4613W	The range of the number of tabs is 0-32.

"Explanation" The count of tabs exceeds the limit.
"Operator response" Change the count of tabs.

E4614W	Please specify the macro name.

"Explanation" The macro name is not specified.
"Operator response" Specify the macro name.

E4615W	Please specify the include path.

"Explanation" The include path is not specified.
"Operator response" Specify the include path.

E4616W	Already a registered macro name. Do you change contents?

"Explanation" The specified macro name has been already existent.
"Operator response" To change the contents, click the OK button.

E4620W	Please specify the start address.

"Explanation" The start address is not found.
"Operator response" Specify the start address.

E4621W	Please specify the end address.

"Explanation" The end address is not found.
"Operator response" Specify the end address.

E4622W	The start address is larger than the end address.

"Explanation" The address range is incorrect.
"Operator response" Check the address range.

E4623W	Please specify a correct start address.

"Explanation" The start address is incorrect.
"Operator response" Specify the correct start address.

E4624W	Please specify a correct end address.

"Explanation" The end address is incorrect.
"Operator response" Specify the correct end address.

E4625W	Please specify the ROM/RAM area name.

"Explanation" The ROM/RAM area name is not specified.
"Operator response" Specify the ROM/RAM area name.

E4626W	Please specify the section name.

"Explanation" The section name is not specified.
"Operator response" Specify the section name.

E4627W	Maximum of address is 0xFFFFFFFF.

"Explanation" The address exceeds the maximum value.
"Operator response" Check the address specification.

E4628W	Maximum of address is 0xFFFFFF.

"Explanation" The address exceeds the maximum value.
"Operator response" Check the address specification.

E4629W	Maximum of address is 0xFFFF.

"Explanation" The address exceeds the maximum value.
"Operator response" Check the address specification.

E4630W	Cannot specify address over bank.

"Explanation" The specified address crosses several banks.
"Operator response" Specify an address within one bank.

E4631W	Specify symbol name.

"Explanation" The symbol name is not found.
"Operator response" Specify the symbol name.

E4632W	Specify set value.

"Explanation" The set value is not found.
"Operator response" Specify the set value.

E4635W	This symbol name has already been registered. Change the setting?

"Explanation" The specified symbol name has been already existent.
"Operator response" To change the setting, click the OK button.

E4636W	This ROM/RAM area name has already been registered. Change the setting?

"Explanation" The specified ROM/RAM area name has been already saved.
"Operator response" To change the setting, click the OK button.

E4637W	This section name has already been registered.Change the setting?

"Explanation" The specified section name has been already existent.
"Operator response" To change the setting, click the OK button.

E4638W	Address must be specified to leader section name.

"Explanation" The address is not specified in the leading section name.
"Operator response" Specify the address.

E4639W	This section name has already been specified in another ROM/RAM area.

"Explanation" The specified ROM/RAM area name has been already existent.
"Operator response" Check the ROM/RAM area name.

E4640W	Specify exact address.

"Explanation" The address specification is incorrect.
"Operator response" Specify the correct address.

E4701W	Specified directory does not exist. Specify?

"Explanation" A non-existent directory is specified.
"Operator response" If there is no error, click the OK button.

E4702W	Cannot specify multiple directories.

"Explanation" Only one directory can be specified.
"Operator response" Specify only one directory.

E4703W	Illegal Environment Variable.

"Explanation" The set value is illegal.
"Operator response" Check the set value.

E4740W	This executable file does not exist. Register in the list?

"Explanation" The file in the execution file name cannot be found.
"Operator response" Check the file name.

E4741W	Title is not specified.

"Explanation" The title is not specified.
"Operator response" Specify the title.

E4742W	Executable file is not specified.

"Explanation" An execution file name is not specified.
"Operator response" Specify an execution file name.

E4743W	The registration count is maximum. You cannot register any more.

"Explanation" No more settings can be saved.
"Operator response" Delete unnecessary settings.

E4744W	Syntax error. Illegal macro is specified.

"Explanation" An undefined option and macro description are found in the execution directory.
"Operator response" Check the syntax.

E4745W	Title is too long.

"Explanation" The title is too long.
"Operator response" Shorten the title.

E4746W	Execute file name is too long.

"Explanation" The execution file name is too long.
"Operator response" Shorten the file name.

E4747W	Option too long.

"Explanation" The specified option is too long.
"Operator response" Shorten the option.

E4748W	The executing directory too long.

"Explanation" The directory name is too long.
"Operator response" Shorten the directory name.

E4749W	Directory not found. Create this directory?

"Explanation" The specified directory cannot be found.
"Operator response" If the directory is correct, click the OK button.

E4750W	Already registered title. Do you change contents?

"Explanation" The specified title has been already existent.
"Operator response" To change the setting, click the OK button.

E4752W	Start tool does not exist.

"Explanation" The tool to be started cannot be found.
"Operator response" Check the existent tool name and directory name.

E4760W	The registered error syntax format cannot be converted.

"Explanation" The error message in the output window cannot be analyzed.
"Operator response" Check the setting in the syntax list in [Setup]-[Set Error Jump].

E4761W	Syntax error. Undefined Macro.

"Explanation" An undefined macro is specified.
"Operator response" Check the syntax.

E4762W	Syntax error. Undefined separate of '%f', '%*'.

"Explanation" The delimiter indicating the end of %f and %* is not input.

"Operator response" The description of the macros, %f and %*, needs the delimiter to identify the end of %f and %*. The next character in the macro description is regarded as the delimiter.

E4763W	Syntax error. Duplicate Macro syntax.

"Explanation" The macros, %f, %l, and %h, are duplicated.

"Operator response" Check the syntax.

E4764W	Syntax error. Invalid '\' syntax.

"Explanation" Invalid| '\' syntax is used for other than \t, \], and \\.

"Operator response" Check the syntax.

E4765W	Syntax error. Invalid '%[]' syntax.

"Explanation" The description of the macro, %[], is illegal.

"Operator response" There may be no correspondence in []. Check the syntax.

E4766W	Syntax error. Don't describe '%f'.

"Explanation" The macro, %f or %h, is not described.

"Operator response" Specify %f or %h in the error jump setting syntax.

E4767W	Syntax error. Invalid Macro into '%[...]'.

"Explanation" An illegal macro is described in the macro, %[].

"Operator response" Only the macro, %% or %] can be described in the macro, %[].

E4768W	Already a registered syntax. Do you change contents?

"Explanation" The same syntax has been already saved.

"Operator response" To change the contents, click the OK button.

E4769W	Syntax not specified.

"Explanation" The syntax is not found.
"Operator response" Specify the syntax.

E4771W	Syntax too long.

"Explanation" The character string in the syntax is too long.
"Operator response" Shorten the syntax.

E4772W	Comment too long.

"Explanation" The comment is too long.
"Operator response" Shorten the comment.

E4773W	The registration count is maximum. You cannot register any more.

"Explanation" The count of existent settings exceeds the maximum value.
"Operator response" Check unnecessary settings.

E4774W	The same syntax has already been set in the SYSTEM. It cannot be changed.

"Explanation" The same syntax has been already set in the SYSTEM.
"Operator response" Syntax that has been already existent in the SYSTEM cannot be changed.

E4780W	Title not specified.

"Explanation" The title is not found.
"Operator response" Specify the title.

E4781W	Execute filename not specified.

"Explanation" The execution file name is not specified.
"Operator response" Specify the execution file name.

E4782W	Option not specified.

"Explanation" The option is not specified.
"Operator response" Specify the option.

E4783W	Already a registered title. Do you change contents?

"Explanation" The specified title has been already existent.
"Operator response" To change the setting, click the OK button.

E4784W	Syntax error. Undefined Macro.

"Explanation" An undefined macro is specified.
"Operator response" Check the syntax.

E4785W	Syntax error. Duplicate Macro syntax.

"Explanation" The macros, %f, %l, and %h, are duplicated.
"Operator response" Check the syntax.

E4786W	Syntax error. Don't describe '%f' or '%h'.

"Explanation" The macro, %f or %h, is not described.
"Operator response" Specify %f or %h in the error jump setting syntax.

E4789W	The registration count is maximum. You cannot register any more.

"Explanation" The count of existent settings exceeds the maximum value.
"Operator response" Delete unnecessary settings.

E4790W	Editor in list not selected.

"Explanation" The editor to be operated is not specified.
"Operator response" Select the required editor from the editor list and operate it.

E4791W	The standard editor cannot delete and change.

"Explanation" An attempt was made to delete or change the standard editor.

"Operator response" The standard editor is built into SOFTUNE Workbench. It cannot be deleted or changed.

E4792W	This executable file does not exist. Register in the list?

"Explanation" The specified execution file cannot be found.

"Operator response" If the execution file name or directory name has no error, register it as it is.

E4793W	The valid editor cannot delete.

"Explanation" An attempt was made to delete the editor selected as the "editor to be used."

"Operator response" Change the "editor to be used" to another before deleting it.

E4794W	Directory not found. Create this directory?

"Explanation" The specified director cannot be found.

"Operator response" To create a directory, click the OK button.

E4795W	Title too long.

"Explanation" The title exceeds the maximum count of characters.

"Operator response" Shorten the title.

E4796W	Execute file name too long.

"Explanation" The execution file name is too long.

"Operator response" Shorten the execution file name.

E4797W	Option string too long.

"Explanation" The option string exceeds the maximum count of characters.

"Operator response" Shorten the option string.

E4798W	The executing directory too long.

"Explanation" The directory name is too long.
"Operator response" Shorten the directory name.

APPENDIX B Error Message for Debuggers

This appendix describes the Error Message for Debuggers

- Error Message for Debuggers

F9201S	Invalid setup file (not found).

"Explanation" The specified setup file could not be found.

"Operator response" Check if the file specified in the startup option setup file specification exists.

F9202S	Invalid command or parameter (in setup file).

"Explanation" An invalid command or parameter exists in the setup file.
Or, the MCU setting information was changed, so it is necessary to change the command or parameter.

"Operator response" Restart the debugger by using the Setup Wizard.

F9203S	Invalid machine program (execution error).

"Explanation" The machine program is already executed or it cannot be executed because the system resources are insufficient.

"Operator response" Check the execution state of the machine program. If the machine program is not executed, close the View Window or terminate another startup program.

E9204S	The disk space is insufficient.

"Explanation" During logging, the free disk space of the storage destination disk is less than 500MB.

"Operator response" Increase disk space of the disk.

E9205S	The Boot-ROM file was not found. Please check the specified folder. "Boot ROM file name"

"Explanation" The Boot ROM file of MCU used was not found.

"Operator response" Check whether the Boot ROM file exists in the folder specified by [Boot ROM] category of [Debug] tab in the setting dialog of the project displayed by [project]-[setting of project] menu.

F9401S	Invalid emulation pod or MCU cable (unmatch or no- connected).

"Explanation" The emulation pod or the MCU cable is not for this product. Alternatively, the MCU cable is not connected correctly.

"Operator response" Turn off the emulator, then check the emulation pod and MCU cable. If the cable is connected normally, then restart the SOFTUNE Workbench.

F9402S	Invalid emulator hardware monitor program (unmatch).

"Explanation" The monitor program loaded into the emulator is not this product.

"Operator response" Start the loader program attached to this product to load the monitor program, then restart the SOFTUNE debugger. For details, refer to "1.1.1 Version Information" in "SOFTUNE Workbench Installation Manual (Release Note)".

F9403S	Emulator hardware error. "detailed information"

"Explanation" The emulator hardware cannot operate normally. Or, processing is stopped because an exception in detailed information is detected.

"Operator response"

- "RAM checker overflow"

  Log data cannot be obtained at the specified interval due to the effect of other applications, etc., and so logging by the RAM Checker is stopped. During logging, do not perform operation that burdens the machine.

- No detailed information is displayed or ERRID is displayed

  Check whether MCU operates normally. Reset and restart the emulator. If the problem occurs frequently, the emulator hardware, MCU, or target system may be down.

F9404S	Invalid emulator hardware monitor program version (old).

"Explanation" The monitor program version in the emulator hardware is old, so the system cannot be normally operated.

"Operator response" Use the monitor loader program to load the monitor attached to this product. For details, refer to "1.1.1 Version Information" in "SOFTUNE Workbench Installation Manual (Release Note)".

[MB2100-01] Enable the automatic loading of the monitor program with the setup wizard.

F9405S	A bus error occurs. To issue reset is necessary for restore.

"Explanation" When the emulator accessed the memory, the bus error occurred.

"Operator response" As the bus error occurred, the emulator cannot control CPU. Please issue reset to restore.

F9406S	Invalid either chip classification in project or chip on board.

"Explanation" The CPU on board is different from the kind of CPU in the project file.

"Operator response" Check the CPU on board or the kind of CPU in the project file.

F9407S	Cannot recognize a communication device and finish a debugger.

"Explanation" Cannot recognize a communication device connecting with the debugger. And terminate the debugger.

"Operator response" Check the connection between PC and emulator.

F9408S	A consistency error. Please turn off the power immediately and check the connection.

"Explanation" The connected emulator does not match emulator defined in the setup file.

"Operator response" Change the emulator type in the setup file on the setup wizard, or connect a proper emulator.

F9409S	Emulator hardware error. Please turn off the power immediately and check the connection.

"Explanation" The emulator hardware cannot operate normally.

"Operator response" Turn off the power immediately and check whether the MCU is operating normally. If this error occurs frequently, the emulator hardware, MCU, and/or target system may be faulty.

F9410S	A configuration board is not connected.

"Explanation" The configuration board is not correctly connected.

"Operator response" Turn off the power supply of the emulator, check the configuration board, connects it normally, and then restart.

F9411S	Cannot continue debugging due to a CR trimming value error.

"Explanation" The preset CR trimming value is abnormal.
"Operator response" Please restart the debug system or change the target device.

F9412S	Invalid supply voltage.

"Explanation" The supply voltage supplied from the user system is found abnormal.
"Operator response" Review the supply voltage of the user system.

F9413S	MB2100-01 is not connected.

"Explanation" USB is not connected or the power supply in MB2100-01 is turned off.
"Operator response" Connect MB2100-01 or turn on the power supply of MB2100-01.

F9414S	DLL version is different. "file name"

"Explanation" The version of the DLL file specified by "File name" is not correct.
"Operator response" Install latest SOFTUNE Workbench.

F9601S	Invalid communication status (or cable connection).

"Explanation" The state of communication line is abnormal, or the cable is not connected correctly.
"Operator response" Check the state of communication line.
Then terminate debugging, turn the emulator off, and turn the emulator on again.
When the USB cable is pulled out during debugging, this message appears.

F9602S	Invalid communication device name (or not specified).

"Explanation" The specified communication device name is incorrect.
"Operator response" Check the communication device name in install file.

F9603S	Invalid INTERFACE (not specified in install file).

"Explanation" INTERFACE is not specified in install file.
"Operator response" Check the install file.

F9604S	Cannot initialize "WINSOCK.DLL".

"Explanation" "WINSOCK.DLL" cannot be initialized.

"Operator response" Check if the TCP/IP protocol is installed on your Windows PC.
If it is not installed, refer to Windows Help to install it.

F9901S	Memory allocation error.

"Explanation" Because of a lack of the memory capacity in host PC, the debug operation cannot be continued.

"Operator response" Terminate the SOFTUNE Workbench, expand an empty memory in host PC, and then restart the SOFTUNE Workbench.

F9902S	System error.

"Explanation" This program could not startup normally because of system error.

"Operator response" Restart the system and then restart the SOFTUNE Workbench.

F9903S	A necessary DLL file was not found.

"Explanation" The required DDL file cannot be loaded.

"Operator response" Re-install SOFTUNE Workbench.

F9904S	The version of CPU information file is an old version.

"Explanation" The version of the CPU information file is old, so information cannot be set properly.

"Operator response" Update the CPU information file to the latest version.

F9905S	A necessary file for USB communication is not found.

"Explanation" A necessary dll-file for the USB communication is not found.

"Operator response" Re-install a SOFTUNE Workbench.

F9906S	Failed in connection because BGM adaptor is too old.

"Explanation" The BGM adapter is too old a version to be connected to the MCU board.

"Operator response" Check the versions of the BGM adapter and MCU board, and connect the acceptable combination.

F9907S	Password is not correct. Please input the correct password by setup wizard.

"Explanation" The password required for starting the debugger is wrong.

"Operator response" Use Setup Wizard to enter the correct password for starting the debugger.

W1001S	Invalid data value (underflow).

"Explanation" Data underflowed the specified precision.

"Operator response" Recheck the precision or data.

W1002S	Invalid data value (overflow).

"Explanation" Data overflowed the specified precision.

"Operator response" Recheck the precision or data.

W1101S	Invalid symbol (multiple).

"Explanation" Duplicate symbols are found.

"Operator response" Recheck the source file corresponding to the load module.

W1102S	Invalid code section or entry data (not found in load module).

"Explanation" The code section and input data are not in the loaded load module. The program counter (PC) is not set.

"Operator response" Set the program counter (PC) and then execute the program.

W1103S	Command history buffer allocation error (in host memory).

"Explanation" Buffer memory for the command history cannot be allocated to an internal memory area in the host machine.

"Operator response" Expand the internal memory area in the host machine. If the Softune debugger is used as is, the command history function cannot be used.

W1104S	Invalid address (mis-alignment).

"Explanation" In the FR family MCU, 16-bit data must be accessed on a 16-bit boundary and 32-bit data on a 32-bit boundary, respectively.

"Operator response" Review the specified address.

W1201S	Invalid HELP command file (not found).

"Explanation" The HELP command file is not placed in a correct location.

"Operator response" Place the HELP command file in a correct location.

W1202S	Loaded different series's file.

"Explanation" The load module file of the MCU series that is different from the target MCU (specified by the project) was loaded.

The instruction operation may not be executed normally in case of the invalid load module.

"Operator response" Do the reload after the file is confirmed, when the specified load module file is not a file of the purpose.

W1203S	Invalid file format.

"Explanation" The file format tried to be loaded is different.

"Operator response" Check the file contents.

W1204S	Loaded different series's file. (FPU instructions are included)

"Explanation" The load module file made with the tool of a series different from the chip specification of the installation file was loaded (FPU instructions are included). In the load module of a different MCU series, the instruction might not operate correctly.

"Operator response" Do the reload after the file is confirmed when the specified load module file is not a file of the purpose.

W1401S	Invalid timer (overflow).

"Explanation" The execution-time timer overflowed during program execution.

"Operator response" Shorten the measurement time.

W1402S	Invalid performance measuring data (buffer full).

"Explanation" The buffer that stores performance measuring data became full during program execution. Performance is not subsequently measured.

"Operator response" Reduce the measurement count.

W1403S	Invalid pass count (overflow).

"Explanation" The pass count overflowed.

"Operator response" Check the term in the expression, then re-enter the command.

W1404S	User reset.

"Explanation" An user reset is specified in MCU during command execution.

"Operator response" Re-enter the [Debug] - [Run] - [Go] menu.

W1405S	The CR trimming value was adjusted due to a CR trimming value error. To continue debugging, adjust the CR trimming value.

"Explanation" The CR trimming value was temporarily adjusted because the preset CR trimming value was abnormal. To continue debugging without encountering the same problem, the CR trimming value of the system has to be adjusted again.

"Operator response" To continue debugging without encountering the same problem, the CR trimming value has to be adjusted. Regarding the adjusting method, please refer to the support web page or contact our support center. After the preset CR trimming value has been adjusted, the debug can be continued; however, Fujitsu cannot guarantee the proper operation of the same device as a mass production device.

W1406S	Command error (Now MCU stopping).

"Explanation" The command that cannot be executed during MCU stopping has been issued.

"Operator response" Check the command.

W1407S	Do not access to debug resource.

"Explanation" When DMAC accesses the area (H'10000..H'103FF) of debugging during the break in the user program, read and write operations cannot be normally processed.

"Operator response" Prohibit the DMA transfer at the break when there is a possibility that DMAC accesses the area of the debug resource.

W1408S	Enabled the software break. The software break is written as an instruction code in the memory.

"Explanation" If the software break is set in the FLASH area, the content in the memory of the setting address is temporarily rewritten.

"Operator response" None

W1601S	Failed to switch to the high-speed communication of DEBUG I/F.

"Explanation" Because a high-speed communication can not be done, the setting of the high-speed communication was changed to the invalidity and switched to the normal communication mode.

"Operator response" Confirm the operating frequency under the setting and make the high-speed communication effective again.

W1901S	The setup file is read-only. The change in setup information cannot be preserved.

"Explanation" The setup file is read-only. Changes to the setup information cannot be saved.

"Operator response" Remove the read-only attribute from the attributes for the setup file corresponding to the setup file name.

W1902S	Invalid CPU information data.

"Explanation" Data in the CPU information file is invalid.

"Operator response" Obtain the latest CPU information file.

W1903S	There is a possibility with an old version of DLL.

"Explanation" The version of the program does not match that of the DLL file.

"Operator response" Install the latest SOFTUNE Workbench.

W1904S	Start "Setup Wizard" to update the setup file.

"Explanation" It is necessary to update information in the setup file, because the information was modified.

"Operator response" Update the setup file on the setup wizard.

W1905S	Please check! There is watch-variable that the realtime-monitoring function becomes invalid.

"Explanation" There is a watch variable that a realtime-monitoring function becomes invalid because a real-time area was changed.

"Operator response" Check the setting of a real-time area and the watch variable.

W1906S	Information file is not the latest. "Information file name"

"Explanation" Information file is not the latest.

"Operator response" Install the latest SOFTUNE Workbench.

E4001S	Command error.

"Explanation" The command, description format or line assembler syntax is incorrect.

"Operator response" Check the command, description format and parameters, then re-enter the command.

E4002S	Command qualifier error.

"Explanation" The specified command qualifier is incorrect or it does not exist in the command.

"Operator response" Check the command qualifier, then re-enter the command.

E4003S	Syntax error.

"Explanation" An error is found in the command or line assembler syntax.

"Operator response" Check the syntax and parameters and then re-enter the parameter.

E4004S	Invalid parameter count (over limit).

"Explanation" The parameter count is too large.
"Operator response" Check the command syntax and then re-enter the parameter.

E4005S	Invalid parameter omission.

"Explanation" A no-omitted parameter is omitted.
"Operator response" Check the command syntax and then re-enter the parameter.

E4006S	Parameter error.

"Explanation" Illegal parameters are specified. The parameter name is illegal or parameters cannot be recognized as numeric values.
"Operator response" Check the command syntax or input radix and then re-enter the parameter.

E4007S	Invalid operand.

"Explanation" There are invalid operands in the expression. Attempts were made to perform arithmetic operations using floating-point numbers. Arithmetic operations using floating-point numbers cannot be performed.
"Operator response" Check the operands in the statement and then re-enter the operands.

E4008S	Invalid operator.

"Explanation" There are invalid operators in the expression.
"Operator response" Check the operators in the expression and then re-enter the operators.

E4009S	Syntax error (operand not found).

"Explanation" The operand is not found in the polynomial operator in the expression.
"Operator response" Check the expression and then input the operand correctly.

E4010S	Syntax error (' " ' or ' ' ' not found).

"Explanation" ' " or ' " on the right side of ' " or ' " is not found in the expression, and the character constant is consisted of one or more characters.
"Operator response" Check the expression and then input quotation marks correctly.

E4011S	Invalid nest level (over limit).

"Explanation" The nest level of ( ), *, and [ ] in the expression exceeds 16. Or, the nest level of the structure or union exceeds 16.

"Operator response" Simplify the expression.

E4012S	Syntax error (dividing by zero).

"Explanation" Division by 0 is found in the expression.

"Operator response" Check the operand in the expression and then re-enter the parameter.

E4013S	Invalid address specifying.

"Explanation" The ending address may be less than the starting address or the specified address range may extend over multiple areas.

"Operator response" Check the addresses, then re-enter the command.

E4014S	Invalid bit pattern (over 0x01 to 0xff).

"Explanation" The value of the specified bit pattern is other than 0x01 to 0xff.

"Operator response" Check the bit pattern and then re-enter the value.

E4015S	Invalid bit offset (over 0 to 31).

"Explanation" The specified bit offset is other than 0 to 31.

"Operator response" Check the bit offset, then re-enter the command.

E4016S	Invalid register or flag name (not found).

"Explanation" The specified register name or flag name is not found.

"Operator response" Check the register name or flag name and then re-enter.

E4017S	Invalid symbol (not found).

"Explanation" The specified symbol is not found in the symbol table. Or, the specified symbol is a local variable and the symbol path name is not registered in the current scope.

"Operator response" Check whether the invalid symbol name is specified or whether the symbol data in the module to which the symbol belongs is registered in the symbol table, and then re-enter. If the symbol data in the module to which the symbol belongs is registered in the symbol table, specify the data with the symbol path name assigned, or register the symbol path name in the current scope.

E4018S	Invalid command alias (not found).

"Explanation" The specified command alias does not exist.

"Operator response" Check the command alias, then re-enter the command.

E4019S	Invalid line number (not found).

"Explanation" The specified line number is not found in the source file. Or, the load module file (line number data) corresponding to the source file is not loaded.

"Operator response" Check the source file and then re-enter. Or, load the load module file corresponding to the source file.

E4020S	Invalid starting display line number (over ending line number).

"Explanation" The source line start line number is larger than the display end line number.

"Operator response" Check the line number and then re-enter.

E4021S	Invalid cycle count (0).

"Explanation" 0 was specified as the cycle count.

"Operator response" Check the cycle count, then re-enter the correct cycle count.

E4022S	Invalid break point number (not found).

"Explanation" The specified break point number is not found.

"Operator response" Check the break point number.

E4023S	Invalid data break point number (not found).

"Explanation" The specified data break point number is not found.
"Operator response" Check the data break point number.

E4024S	Invalid watch point number (not found).

"Explanation" The specified watch point number is not found.
"Operator response" Check the watch point number.

E4025S	Invalid starting display trace number (over ending number).

"Explanation" The starting display trace number is larger than the display ending trace number.
"Operator response" Check the trace number, then re-enter the number.

E4026S	Invalid format statement characters.

"Explanation" The specified format statement character string is incorrect.
"Operator response" Check the format statement character string, then re-enter the command.

E4027S	Invalid symbol (not found) path name.

"Explanation" The specified symbol path name is not found.
"Operator response" Check the symbol path name and then re-enter the name.

E4028S	Invalid function (not found, or argument error).

"Explanation" The specified function is not found. Or, the invalid argument of the function is specified.
"Operator response" Check the function or argument and then re-enter it.

E4029S	Invalid expression (used variable of structure or union type).

"Explanation" The structure or union variable cannot be used as the operand in the language expression.
"Operator response" Recheck the data format. Prefix the operator & to the variable.

E4030S	Invalid address (not found).

"Explanation" The corresponding address is not found in the line number.

"Operator response" Recheck the line number.

E4031S	Invalid automatic variable reference.

"Explanation" Attempts are made to refer the automatic variable out of the function in which the variable is defined.

"Operator response" The automatic variable can be referred only within the function in which the variable is defined.

E4032S	Invalid variable specifying.

"Explanation" The specified variable is not the member of the structure or union variable.

"Operator response" Check the structure or union member.

E4033S	Floating point data format error.

"Explanation" The floating-point data format is illegal.

"Operator response" Recheck the floating-point data format.

E4034S	Invalid macro command definition (not found).

"Explanation" The specified macro command name is not found.

"Operator response" Check the macro command name, then re-enter the command.

E4035S	Invalid symbol address (not found).

"Explanation" The address of a register variable or address of a variable assigned to a register is referenced.

Example:

```
SET DATABREAK &r
```

If the register assigned variables exists in the address space of CPU, the address can be referenced.

"Operator response" The address of a register variable or address of a variable assigned to a register cannot be referenced.

E4101S	Invalid command list nest level (over 8).

"Explanation" The nesting level of command list of the command procedure, command macro, or break point exceeds 8.

"Operator response" Review the execution of the command.

E4102S	Symbol definition error.

"Explanation" The free area allocated in host machine memory is insufficient to execute commands. This error occurs when too many device drivers are incorporated under the MS-DOS (PC) environment.

"Operator response" Expand the free area allocated in host machine, then restart the SOFTUNE Workbench.

E4103S	OS command error.

"Explanation" An OS command cannot be executed. The command shell format is incorrect.

"Operator response" Start the command shell of correct format.

E4104S	Invalid command shell (not found).

"Explanation" The command shell could not be found.

"Operator response" Review the environment variable, etc., so that the command shell can be started.

E4105S	Invalid alias string.

"Explanation" The command alias includes an unregisterable character.

"Operator response" Review command alias registration, then re-enter the command.

E4106S	Invalid macro command name (registered already).

"Explanation" The same macro command is already registered.

"Operator response" Review the macro command name, then re-enter the command.

E4107S	Invalid memory map definition.

"Explanation" Memory mapping is too complex to define the area. When setting the memory area attributes, the areas with different attributes are excessive, causing the internal table to overflow.

"Operator response" Simplify the memory mapping.

E4108S	Memory allocation error.

"Explanation" There is insufficient memory space for command execution by the host machine memory. This error occurs when too many softwares are executed.

"Operator response" Increase the memory space in the host machine to restart the SOFTUNE Workbench.

E4109S	Object loading error.

"Explanation" The object load destination exceeds the maximum value of address.

"Operator response" Check the object size and object load destination or the specified address.

E4110S	Log file open error (already).

"Explanation" The log file is already open.

"Operator response" Close the current log file, then open a new log file.

E4111S	Memory access error.

"Explanation" Attempts were made to access undefined memory. The address where access causing an error is made is displayed in the address part.

"Operator response" Check the memory mapping.

E4112S	Invalid nest level of structure or union (over 16).

"Explanation" The debug data table could not be created in the host machine memory.

"Operator response" Increase the memory space in the host machine, then restart the SOFTUNE Workbench.

E4113S	Debug data table creation error.

"Explanation" The debug data table cannot be created in memory of the host machine or in the directory specified in TMP in the install file.

"Operator response" Increase the memory space in the host machine and restart the SOFTUNE Workbench. Or, check the condition of the directory specified in TMP in the install file.

E4114S	Logging control command error.

"Explanation" The log file was operated, although it is not open.

"Operator response" Check whether the log file is open.

E4115S	Invalid alias name (registered already).

"Explanation" The same command alias is already registered.

"Operator response" Review the command alias, then re-enter the command.

E4116S	Invalid alias name (not found).

"Explanation" The specified command alias does not exist.

"Operator response" Check the command alias, then re-enter the command.

E4117S	Data type error.

"Explanation" The data type is unmatched.

"Operator response" Check the data type and then re-enter the data type.

E4118S	Invalid member name (not specified).

"Explanation" The structure or union name cannot be specified.

"Operator response" Specify the member name.

E4119S	Break point and data break point setting error.

"Explanation" Break points and data break points cannot be set. When the data monitoring break cannot set in the MB2198 or MB2147-01 emulator, display the message.

"Operator response" Check the contents of break points and maximum count of break points. In case of MB2198 or MB2147-01, the break point, the trace trigger and the data watch break may be used same hardware point. Check the status in event lists. In case of MB2145-506, MB2145-507 emulator pod or MB2147-05 emulator, the specified break point be set at address out of debug area and break points exceeds maximum count. Check the debug area.

E4120S	CALL command error.

"Explanation" The CALL command is already executing; it cannot be nested.

"Operator response" Suspend the CALL command with a CLEAR CALL command. Alternatively, execute the GO or STEP command until the call operation terminates, then execute the CALL command.

E4121S	Invalid function (at the top).

"Explanation" There is no higher-level function than this function or this function is called from a program other than a C program.

"Operator response" Check the current function.

E4122S	Invalid function (at the bottom).

"Explanation" There is no lower-level function than this function or this function is called from a program other than a C program.

"Operator response" Check the current function.

E4123S	Invalid coverage map (over-full).

"Explanation" The coverage area cannot be set any more.

"Operator response" Simplify coverage area specification.

E4124S	Coverage area setting error.

"Explanation" The coverage area is not set.

"Operator response" Set the coverage area.

E4125S	Invalid coverage area.

"Explanation" An area outside the coverage area was specified.
"Operator response" Check and specify the coverage area.

E4126S	Invalid coverage file.

"Explanation" A file other than the coverage file was specified.
"Operator response" Check the file data.

E4127S	Invalid debug data (not loaded).

"Explanation" The debug data file has not been loaded.
"Operator response" Load the debug data file, then specify a coverage.

E4128S	Mapping overlap.

"Explanation" The specified map area overlaps another area.
"Operator response" Check the map specification, then re-enter the command.

E4129S	Invalid address (mis-alignment).

"Explanation" In the FR family MCU, 16-bit data must be aligned on a 16-bit boundary and 32-bit data on a 32-bit boundary, respectively.
"Operator response" Review the specified address.

E4130S	Cannot open current source window.

"Explanation" The source window that displays the current location could not be found in the set source search directory.
"Operator response" Set the directory containing the source file.

E4131S	Cannot be used in current mode of debugger.

"Explanation" The functions that can be used depend on the type of the debugger.
"Operator response" Check the type of debugger.

E4132S	Command error (debug mode violation).

"Explanation" The function that can be used varies with the debug function.

"Operator response" Check the setting of debug function by selecting [Environment] - [Debug environment] - [Select Debug function] menu.

E4133S	Task debugging cannot be used.

"Explanation" The task debug function cannot be used if an available object with that function is not loaded.

"Operator response" Load the object with the task debug function.

E4134S	MMU data access exception.

"Explanation" An MMU access error occurs.

"Operator response" Check whether invalid memory was accessed.

E4135S	Register exception.

"Explanation" An undefined register is accessed.

"Operator response" Turn on the target power supply again if recovery fails. Check whether a valid register is accessed.

E4136S	Not the first address of a parallel instruction.

"Explanation" The specified address is not the first an address of the instruction executed in parallel.

"Operator response" Specify the first address of the instruction executed in parallel.

E4137S	Register access error.

"Explanation" Cannot access the register.

"Operator response". There is an inaccessible register according to the kind or the state of the chip. Please refer to the hardware manual of the product used for details.

E4138S	Invalid combination of size and address beyond 64 bits boundary.

"Explanation" Cannot specify the address beyond 64 bits boundary.

"Operator response" Please specify the address matched 64 bits boundary.

E4139S	Over point.

"Explanation" The set point count is above maximum.
This message appears when the trace trigger or the data watch break and RAM Checker is set with for MB2198 for FR and MB2147-01 for F^2MC-16 emulator.

"Operator response" Please check the current set point count and state, then delete the unnecessary points and set the point again.

E4140S	Invalid point number.

"Explanation" Cannot find the specified point number.
If the invalid point number is specified (e.g. deleting event point), this message appears.

"Operator response" Please check the point number.

E4141S	The same address is set.

"Explanation" It is duplicate that the function is set at one address.
For example, the same address is specified to the multiple levels on each sequencer, this message appears.

"Operator response" Please check the current set point count and state, then delete the unnecessary points and set the point again.

E4142S	Invalid break condition.

"Explanation" The data break condition is not set with MB2198 emulator for FR.

"Operator response" Please check the current set point count and state, then delete the unnecessary points and set the point again.

E4143S	No supported function with this device. (MBXXXX)

"Explanation" No supported function with this device was specified.

"Operator response" This message appears when the evaluation chip or the emulator does not have the specified function. Please see the hardware manual of the product used or "SOFTUNE Workbench Command Reference Manual" based on MBXXXXXX in the error message.

E4144S	Please use the RAM checker function with an USB device.

"Explanation" The RAM Checker cannot be used because other than USB (RS or LAN) is used as the communication device.

"Operator response" When using the RAM Checker, use USB as the communication device.

E4201S	File access error.

"Explanation" The file cannot be accessed.

"Operator response" Check the condition of the disk in the host machine.

E4202S	File close error.

"Explanation" The file cannot be closed.

"Operator response" Check the condition of the disk in the host machine.

E4203S	File open error.

"Explanation" The file cannot be opened.

"Operator response" Check the file name or the condition of the disk in the host machine. Or, check the file and directory.

E4204S	Data write error.

"Explanation" Data cannot be written to the file.

"Operator response" Check the condition of the disk in the host machine.

E4205S	Invalid line number (not found).

"Explanation" The corresponding source line is not found at the specified address. Even if the corresponding source line is not found, the source line is displayed in the source window.

"Operator response" Review the address. Or, load the load module with debug information.

E4206S	Alias file load error.

"Explanation" The specified alias file cannot be loaded.

"Operator response" Check the alias file name or the disk state of the host machine. Alternatively, check the directory containing the alias file.

E4207S	Alias file save error.

"Explanation" The specified alias file cannot be saved.

"Operator response" Check the condition of the disk in the host machine.

E4208S	Invalid file format.

"Explanation" The format of the file to be loaded is illegal.

"Operator response" Check the file.

E4209S	Open file read error.

"Explanation" An error occurred during reading of the opened file.

"Operator response" Check the file (drive) being read.

E4301S	Invalid interrupt factor number.

"Explanation" The specified interrupt number does not exist.

"Operator response" Specify the interrupt number which can be assigned (IRQ0 to IRQ47).

E4302S	Invalid I/O buffer number.

"Explanation" The specified I/O buffer number does not exist.

"Operator response" The simulator provides 0 to 3 I/O buffers.

E4303S	Invalid port address.

"Explanation" An address was specified beyond the port address range.

"Operator response" A port address can be specified only in the MCU I/O area. Specify an address in the MCU I/O area.

E4304S	Invalid output destination.

"Explanation" A data output destination, which is already in use as the data output destination, was specified.

"Operator response" Specify a data output destination not in use.

E4305S	Invalid port count.

"Explanation" The count of specified ports exceeds 4096.

"Operator response" Specify 4096 ports or less.

E4306S	Simulation memory allocation error.

"Explanation" Simulation memory cannot be allocated to an internal memory area in the host machine.

"Operator response" Expand the internal memory area in the host machine.

E4307S	Invalid inport data file.

"Explanation" The file name assigned to the inport is incorrect or the file does not exist.

"Operator response" Check the general format of the file.

E4401S	Verify error.

"Explanation" A verify error occurred when data was being written to memory by a command.

"Operator response" Check whether that data was written to the I/O area where values change and that memory is mounted. Also check whether or not a memory error occurred.

E4402S	Parity error (at emulation memory).

"Explanation" A parity error occurred at accessing to the emulation memory.

"Operator response" Reset the emulator body, then restart it. If the error occurs frequently, it may be an emulation memory malfunction.

E4403S	Parity error (at debug memory).

"Explanation" A parity error occurred at accessing to the memory for emulator operation.

"Operator response" Reset the emulator body, then restart it. If the error occurs frequently, it may be a malfunction of the memory for emulator operation.

E4404S	Command error (MCU is busy).

"Explanation" An unexecutable command was tried to execute during MCU execution.

"Operator response" Check the command.

E4408S	Invalid event number (not found).

"Explanation" The specified event number is not found.

"Operator response" Check the event number.

E4409S	Invalid level number (not found).

"Explanation" The specified level number is not found.

"Operator response" Check the level number.

E4410S	Command error (event mode violation).

"Explanation" A command was specified that violates the event mode.

"Operator response" Check the event mode setting with the Debug Environment in the Setup menu.

E4411S	Invalid latch number (not found).

"Explanation" The specified latch number is not found.

"Operator response" Check the latch number.

E4412S	Invalid supply voltage.

"Explanation" The supply voltage supplied from the user system is found abnormal.

"Operator response" Review the supply voltage of the user system.

E4413S	MCU is in STOP mode, or Invalid system clock.

"Explanation" This indicates one of the following.
- MCU is in STOP mode.
- The system clock supplied from the user system is found abnormal.

"Operator response" Review the MCU status or the system clock of the user system.

E4414S	MCU reset error.

"Explanation" The MCU reset cannot be executed normally.

"Operator response" The mode data and the reset vector read at reset may be an incorrect value. Set a correct value and retry this command. When this error occurs if the mode data is read from the user memory, the user memory cannot be read. Therefore, map it in the emulation memory before executing the [Debug] - [Reset MCU] menu.

E4415S	Invalid MCU.

"Explanation" Commands cannot be executed because MCU is not an operational state.

"Operator response"

1. Set the reset vector and the mode data, then execute the [Debug] - [Reset MCU] menu.
2. Release the SLEEP, STOP or HOLD state on the user system side, or set the reset vector and the mode data, then execute the [Debug] - [Reset MCU] menu. Note that the HOLD state cannot be released by the [Debug] - [Reset MCU] menu.
3. Check the execution result of the command.

E4416S	Invalid jump level number.

"Explanation" The jump destination level number of the sequencer is incorrect.

"Operator response" Review the jump destination level number. The sequencer cannot jump to the same level as the level to be specified.

E4417S	Command error (on internal ROM real-time mode).

"Explanation" The command cannot be executed because the MCU execution mode is native.

"Operator response" Change the MCU execution mode to debug.

E4418S	Command error (user reset).

"Explanation" This command cannot be executed because user reset is specified. This error occurs even if user reset is already released.

"Operator response" Release user reset, execute the [Debug] - [Reset MCU] menu, then execute this command.

E4419S	Abort command error.

"Explanation" The ABORT command cannot be executed due to the SLEEP or STOP state.

"Operator response" Release the SLEEP or STOP state.

E4420S	Command error (hardware standby).

"Explanation" This command cannot be executed due to the hardware standby state. This error occurs even if the hardware standby state is already released.

"Operator response" Release the hardware standby state, execute the [Debug] - [Reset MCU] menu, then execute this command.

E4421S	Command error (timer-mode violation).

"Explanation" When the timer mode is "timer", the SHOW CYCLE command and CLEAR CYCLE command cannot be executed. When the timer mode is "cycle", the SHOW TIMER command and CLEAR TIMER command cannot be executed.

"Operator response" Check the timer mode, then re-enter the command.

E4422S	Invalid break point (not found).

"Explanation" The software break point became invalid because data in the address where the software break point is set was rewritten by program execution. Alternatively, the software break point remained in memory because an error occurred when the point was being reburied. In this case, data in the program being loaded and setting data at the software break point are not guaranteed.

"Operator response" Delete all software breaks, then review the program data. If some software breaks still remain in memory, reload the program.

E4423S	Monitor hit stack-check function.

"Explanation" A stack-check exception occurred within the monitor at returning to the user-program.

"Operator response" Invalidate a stack-check function or increase usable stack area.

E4424S	Exception occurred while accessing user resource. ["detailed information"]

"Explanation" Because the exception shown in detailed information had been detected while accessing a specified user resource, processing was interrupted.

"Operator response" Please confirm the corresponding matter from the exception generation factor described in the hardware manual of the product used based on detailed information. In FR-V, the address and trap type (TT) at the exception occurrence are displayed as detailed information in the form of "address TT:nn". Please refer to "4.7.2.3 Debug Environment" of "SOFTUNE Workbench operation manual" for the factor which relates to MMU because the operation by the debugger might be necessary.

E4425S	Invalid area number.

"Explanation" The specified area number is not found.

"Operator response" Specify an area number which is possible to set.

E4426S	Command error (event mode violation <Performance>).

"Explanation" Event mode is set to performance mode, and so the command cannot be executed.

"Operator response" Change event mode to a mode where the command can be used.

E4427S	Command error (event mode violation <Single trace>).

"Explanation" Event mode is set to single trace mode, and so the command cannot be executed.
"Operator response" Change event mode to a mode where the command can be used.

E4428S	Command error (event mode violation <Multi trace>).

"Explanation" Event mode is set to multi trace mode, and so the command cannot be executed.
"Operator response" Change event mode to a mode where the command can be used.

E4429S	Command error (The trace function is operating).

"Explanation" The command that cannot be executed during the trace function has been issued.
"Operator response" Check the command.

E4430S	If access size is word, this event cannot be set. Recommend 'Don't care'.

"Explanation" Though the access size was specified for the word, the odd address is tried to set.
"Operator response" Specify the access size for "Size Free".

E4431S	Data on the RAM is broken.

"Explanation" The recovery of RAM area that the debugger has used temporarily failed. The debugger uses a part of internal RAM under the following conditions.
- When the execution starts
- When the break is executed
- When the reset is executed
- When the debug completes

"Operator response" Set the RAM contents again if necessary.

E4432S	Command error (pass count mode violation).

"Explanation" The command that can not be used in a current pass count mode was input.
"Operator response" Switch the pass count mode.

E4433S	Command error (time measurement mode violation).

"Explanation" The command that can not be used in a current execution time mode was input.

"Operator response" Switch the execution time mode.

E4434S	Detected debug resource access violation (by user operation [program, DMA]).

"Explanation" The access to the debugging resource was detected during the break in the user program. The access to the debugging resource is not correctly processed during the break in the user program.

"Operator response" When there is a possibility that the DMA transfer is done to the debug resource during the break in the user program, [Accepting DMA] in [Emulation] tab of the debug environment setting dialog box is set to the prohibition and do not generate the DMA transfer.

E4435S	Command error (The software break is not allowed).

"Explanation" The software break tried to be set though the setting of the software break was prohibited.

"Operator response" Switch the software break to the use permission.

E4436S	Command error (The performance function is operating).

"Explanation" The command could not be used while the performance measuring tried to be executed.

"Operator response" Confirm the command.

E4437S	The peripheral resources of Sleep mode was accessed.

"Explanation" The peripheral resource in the sleep state cannot be accessed.

"Operator response" Do not access to the peripheral resource in the sleep state.

E4438S	Failed to switch to the high-speed communication of DEBUG I/F.

"Explanation" The current reference clock for a high-speed communication is beyond the limits of the clock that can be set.

"Operator response" Check whether to mistake the set reference clock for the high-speed communication.

E4501S	Verify error.

"Explanation" A verify error occurred when data was being written to memory by a command.

"Operator response" Check that data was written to the I/O area where values change and that memory is mounted. Also check whether or not a memory error occurred.

E4502S	Illegal stack area.

"Explanation" The stack area used by the monitor debugger cannot be accessed.

"Operator response" Secure the correct stack area.

E4503S	System call error (cannot execute).

"Explanation" In this state, a system call cannot be executed normally.

"Operator response" Execute a system call in the state in which system calls can be issued. Interrupts may be disabled.

E4504S	This command is not built-in.

"Explanation" The associated function is not built in a target side.

"Operator response" Built the associated function in the target-side program.

E4505S	FLASH area can be changed only by LOAD command.

"Explanation" Only the LOAD command can be used to access the Flash area.

"Operator response" To update the Flash area, create an update file and load it using the LOAD command.

E4506S	The file of mixed ROM/RAM areas can not be loaded.

"Explanation" The same file cannot be loaded into the ROM and RAM areas.

"Operator response" The file to be loaded into the Flash area and the one to be loaded into the RAM area must be created separately.

E4601S	Invalid communication status (or cable connection).

"Explanation" The communication line state is abnormal or the cable connection is incorrect.

"Operator response" Check the line connection state.

Push the reset button on the target board when CPU runs in the sub-clock mode.

E4602S	Communication: Parallel adapter not connected.

"Explanation" The parallel adapter is not connected.

"Operator response" Connect the parallel adapter to MB2141 correctly, then re-execute this program.

E4603S	Communication: Mismatch parallel adapter version.

"Explanation" Communication cannot be performed because the version of the parallel communication adapter is old.

"Operator response" Use the latest parallel communication adapter.

E4604S	Communication: Cannot find host name.

"Explanation" The specified host name is not registered in the hosts file.

"Operator response" Please register the host name in the hosts file.

For details, refer to the "Appendix C Setting LAN Interface" of "SOFTUNE Workbench Operation Manual".

E4605S	Communication: Cannot find port number.

"Explanation" The port number of ICE is not defined in the services file.

"Operator response" Please register the port number in the services file.

For details, refer to the "Appendix C Setting LAN Interface" of "SOFTUNE Workbench Operation Manual".

E4606S	Communication: Cannot open device.

"Explanation" Abnormality is found in the specified device or not connected correctly.

"Operator response" Please confirm whether the specified device is correctly connected.

E4607S	Communication: Time out.

"Explanation" Reception information on transmission information was not received within the fixed time.

"Operator response" Please confirm whether the specified device is correctly connected.

E4608S	Communication : DEBUG I/F error.

"Explanation" The problem occurred by the communication in DEBUG I/F.

"Operator response" Confirm the connection of the DEBUG I/F cable.

E4609S	Communication : USB error.

"Explanation" The problem occurred by the USB communication.

"Operator response" Confirm the connection of the USB cable.

E4610S	Communication : Time out (DEBUG I/F).

"Explanation" It is not possible to communicate with the target. The possibility that there is a problem in the target is high.

"Operator response" Reactivate the target.

E4611S	Communication : Time out (debug program).

"Explanation" It is not possible to communicate with the target. The possibility that there is a problem in the target is high.

"Operator response" Reactivate the target.

E4901S	Not enough timer resource.

"Explanation" The timer resource of Windows cannot be used.

"Operator response" End other applications, then re-execute this command.

E4902S	The key code cannot be defined.

"Explanation" The key code cannot be defined.

"Operator response" Define another key code.

E4903S	Write error (at flash memory).

"Explanation" During programming to Flash ROM, timing limit over is occurred.
"Operator response" When this error occurs, contact your Fujitsu representative immediately.

E4904S	Erase error (at flash memory).

"Explanation" During erasing to Flash ROM, timing limit over occurred.
"Operator response" When this error occurs, contact your Fujitsu representative immediately.

E4905S	Don't use because enabled MCU security.

"Explanation" When the MCU security was effective, the function that can not be used was operated.
"Operator response" Release the MCU security.

APPENDIX C Execution Suspension Messages List

This appendix describes the Execution Suspension Messages List

- Execution Suspension Messages List

Break at address by break point

"Explanation" This message is displayed when a break is caused by a software break point.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by hardware break point

"Explanation" This message is displayed when a break is caused by a hardware break point (including break point specified by GO command).

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by code event break (No. code-event-number)

"Explanation" This message is displayed when a break is caused by a code event.

Address indicates the address of the next instruction to be executed where execution was suspended.

Code-event-number indicates the number of the code event that caused the break.

Break at address by code event break (sequential)

"Explanation" This message is displayed when a sequential break is caused by code event 1 or 2.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by data event break (No. data-event-number)

"Explanation" This message is displayed when a break is caused by a data event.

Address indicates the address of the next instruction to be executed where execution was suspended.

Data-event-number indicates the number of the data event that caused the break.

Break at address by data event break (sequential)

"Explanation" This message is displayed when a sequential break is caused by data event 1 or 2.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by trace buffer full

"Explanation" This message is displayed when a break is caused by a trace buffer full.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by alignment error break (code)

"Explanation" This message is displayed when a break is caused by a code fetch alignment error.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by alignment error break (data)

"Explanation" This message is displayed when a break is caused by a data access alignment error.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by external trigger break

"Explanation" This message is displayed when a break is caused by the input of an external signal to the TRIG pin of the emulator.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by trace lost break

"Explanation" This message is displayed when a break is caused by the trace data loss.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by data break at access-address

"Explanation" This message is displayed when a break is caused by a data break point.

Address indicates the address of the next instruction to be executed where execution was suspended.

Access-address indicates the address where the access that caused the break was made.

Break at address by guarded access access-type at access-address

"Explanation" This message is displayed when a break is caused by any of the following accesses.
- Code fetch access to a code fetch inhibited area
- Read access to a read-inhibited area
- Write access to a write-inhibited area

There may be an error in the memory attribute or the program.

Address indicates the address of the next instruction to be executed where execution was suspended.

Access-type indicates the type of the access that caused the break.

Access-address indicates the address where the access that caused the break was made.

Break at address by dispatch task from task ID= <dispatch-source-task-ID > to task ID= <dispatch-destination-task-ID >

"Explanation" This message is displayed when a break is caused by task dispatch.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by system call <system-call-name > on task ID= <task-ID >

"Explanation" This message is displayed when a break is caused by a system call.

Address indicates the address of the next instruction to be executed where execution was suspended.

System-call-name indicates the name of the system call that caused the break. Task-ID indicates the ID of the task that issued the system call.

Break at address by command abort request

"Explanation" This message is displayed when a break is caused by the ABORT command on the Debug menu.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by output file overflow

"Explanation" This message is displayed when a break occurs because data could not be written to the data output file of an output port.

Check the data output file of the output port.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by stop abnormal action

"Explanation" This message is displayed when a break occurs because a non-executable instruction was added after a prefix instruction.

Check the program because it may be incorrect.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by invalid call termination

"Explanation" The CALL command is executed after a break point is set in the address indicated by the current PC and the RP register is set so that control will return to the address. For this reason, a break occurs if the address of the original PC is executed during execution of the CALL command.

In this way, this message is displayed when a break occurs before execution of the CALL command is completed.

Restart execution of the CALL command with the GO command as is or suspend execution with the CLEAR CALL command.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by EIT (attached information)

"Explanation" This message is displayed when a break is caused by EIT.

Address indicates the address of the next instruction to be executed where execution was suspended.

For attached information, refer to the hardware manual of the product used.

Break at address by step command

"Explanation" This message is displayed by the SHOW STATUS command when a break is caused by step (INTO) execution.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by call command

"Explanation" This message is displayed when a break occurs after execution of the CALL command is completed.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by violation to combine instructions

"Explanation" Displayed when a combined instruction that is not allowed in 1 parallel instruction is executed and there is a break in the simulator debugger.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by slot issue violation

"Explanation" Displayed when an instruction that cannot be issued to a slot in 1 parallel instruction is executed and there is a break in the simulator debugger.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by resource write-write confliction

"Explanation" Displayed when executing a plurality of write access of the same memory or register of 1 parallel instruction in the simulator debugger.

The address is the one of the parallel instruction executed after the instruction that had the cause of the break.

Break at address by data watch break

"Explanation" Displayed when a break is caused by a data watch break point in the emulator debugger.

Address indicates the address of the break factor instruction.

Break at address by unknown break factor

"Explanation" Displayed when a break is caused by indefinite factor.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by trace end break

"Explanation" Displayed when the break is caused by the break on the completion of trace.

Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by breakpoint (data watch)

"Explanation" Displayed when the break is caused by the data watch break.

Address indicates the address where the command has caused the break.

Break at address by sequential break

"Explanation" This message is displayed when a break is caused by a data event. Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by data event break

"Explanation" Displayed when the break is caused by the sequencer.
Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by sequential or pass count break

"Explanation" Displayed when the break is caused by sequence or hardware/count break.
Address indicates the address of the next instruction to be executed where execution was suspended.

Break at address by guarded access

"Explanation" This message is displayed when a break is caused by code fetch access to a code fetch inhibited area, read access to a read-inhibited area, or write access to a write-inhibited area. There may be an error in the memory attribute or the program. Address indicates the address of the next instruction to be executed where execution was suspended.

Note:

If the CPU pause state is released during execution of the user program, that information is also displayed as additional information.

Example: When stopped due to the forced break:
Break at [address] by command abort request (exit CPU pause)

It should be noted that there are four types of CPU pause state:

Sleep Mode/ Stop Mode/ Ready Mode/ Hardware standby Mode

INDEX

Symbols

%B
%B 229
%BIT
%BIT 229
%CALL
%CALL 222
%D
%D 229
%ENTRY
%ENTRY 224
%ERRNUM
%ERRNUM 223
%EVAL
%EVAL 238
%L
%L 229
%LOADNUM
%LOADNUM 228
%RADIX
%RADIX 226
%S
%S 229
%SCPADR
%SCPADR 227
%STKTOP
%STKTOP 225
%STRCAT
%STRCAT 234
%STRCMP
%STRCMP 232
%STRGET
%STRGET 230
%STRLEN
%STRLEN 233
%STRSTR
%STRSTR 231
%SYMLEN
%SYMLEN 235
%TOSTR
%TOSTR 237
%TOVAL
%TOVAL 236
%W
%W 229

A

ALIAS
CANCEL ALIAS 192
SET ALIAS 190
SHOW ALIAS 191
ASSEMBLE
ASSEMBLE 168

B

BATCH
BATCH 186
BREAK
BREAK 220
CANCEL BREAK 69
DISABLE BREAK 71
ENABLE BREAK 70
SET BREAK (type 1) 61
SET BREAK (type 2) 64
SET BREAK (type 3) 65, 66
SHOW BREAK 67

C

CALL
CALL 54
CLEAR CALL 56
CALLS
SHOW CALLS 122
CANCEL
CANCEL ALIAS 192
CANCEL BREAK 69
CANCEL COVERAGE 116
CANCEL DATABREAK 77
CANCEL EVENT 83
CANCEL INPORT 27
CANCEL INTERRUPT 34
CANCEL LOGGING 203
CANCEL MAP 20
CANCEL OUTPORT 31
CANCEL RAMMONITOR 140
CANCEL SEQUENCE 91
CANCEL VARIABLE 197
CANCEL WATCH 42
CLEAR
CLEAR CALL 56
CLEAR COVERAGE 117
CLEAR MULTITRACE 102
CLEAR TIMER 125
CLEAR TRACE 132

CLEARE
CLEAR PERFORMANCE....111
COPY
COPY 165
COVERAGE
CANCEL COVERAGE....116
CLEAR COVERAGE....117
DISABLE COVERAGE 119
ENABLE COVERAGE 118
SET COVERAGE....112
SHOW COVERAGE....113

D

DATABREAK
CANCEL DATABREAK 77
DISABLE DATABREAK....79
ENABLE DATABREAK....78
SET DATABREAK (type 1) 72
SET DATABREAK (type 2) 74
SHOW DATABREAK 76
DELAY
SET DELAY 95
SHOW DELAY 96
DISABLE
DISABLE BREAK....71
DISABLE COVERAGE 119
DISABLE DATABREAK....79
DISABLE EVENT....85
DISABLE LOGGING 205
DISABLE MULTITRACE....104
DISABLE RAMMONITOR....143
DISABLE SEQUENCE....94
DISABLE TRACE....134
DISABLE VERIFYMODE 22
DISABLE WATCHDOG....38
DISASSEMBLE
DISASSEMBLE 169
DOWN
DOWN....184
DUMP
DUMP 163

E

ENABLE
ENABLE BREAK....70
ENABLE COVERAGE 118
ENABLE DATABREAK....78
ENABLE EVENT....84
ENABLE LOGGING 204
ENABLE MULTITRACE....103
ENABLE RAMMONITOR....142
ENABLE SEQUENCE....93
ENABLE TRACE....133
ENABLE VERIFYMODE 21
ENABLE WATCHDOG 37
ENTER
ENTER....148
EVENT
CANCEL EVENT 83
DISABLE EVENT....85
ENABLE EVENT....84
SET EVENT 80
SHOW EVENT 82
EXAMINE
EXAMINE....146
EXIT
EXIT 4

F

FILL
FILL....161

G

GO
GO 44
SET GO....46
SHOW GO....47

I

IF
IF 216
INITIALIZE
INITIALIZE....3
INPORT
CANCEL INPORT 27
SET INPORT 24
SHOW INPORT....26
INTERRUPT
CANCEL INTERRUPT 34
SET INTERRUPT 32
SHOW INTERRUPT 33

L

LIST
LIST....178
LOAD
LOAD 172
LOGGING
CANCEL LOGGING....203
DISABLE LOGGING....205
ENABLE LOGGING....204
SET LOGGING....200
SHOW LOGGING....202

M

MAP
CANCEL MAP 20
SET MAP (type 1) 16
SET MAP (type 2) 18
SHOW MAP 19
MEMORY
SEARCH MEMORY 155
SET MEMORY 150
SHOW MEMORY 152
MODE
SET MODE 6
SHOW MODE 8
MOVE
MOVE 162
MULTITRACE
CLEAR MULTITRACE 102
DISABLE MULTITRACE 104
ENABLE MULTITRACE 103
SEARCH MULTITRACE 105
SET MULTITRACE 99
SHOW MULTITRACE 100

O

OUTPORT
CANCEL OUTPORT 31
SET OUTPORT 28
SHOW OUTPORT 30
OUTPUT
SET OUTPUT 208
SHOW OUTPUT 209

P

PATH
SET PATH 180
SHOW PATH 181
PERFORMANCE
CLEAR PERFORMANCE 111
SET PERFORMANCE 107
SHOW PERFORMANCE 109
PRINTF
PRINTF 206

Q

QUIT
QUIT 188

R

RADIX
SET RADIX 11
SHOW RADIX 12
RAMMONITOR
CANCEL RAMMONITOR 140
DISABLE RAMMONITOR 143
ENABLE RAMMONITOR 142
SET RAMMONITOR 137
SHOW RAMMONITOR 139
REGISTER
SET REGISTER 157
SHOW REGISTER 158
REPEAT
REPEAT 218
RESET
RESET 5

S

SAMPLING
SET SAMPLING 120
SHOW SAMPLING 121
SAVE
SAVE 175
SCOPE
SHOW SCOPE 182
SEARCH
SEARCH MEMORY 155
SEARCH MULTITRACE 105
SEARCH TRACE 135
SEQUENCE
CANCEL SEQUENCE 91
DISABLE SEQUENCE 94
ENABLE SEQUENCE 93
SET SEQUENCE 86
SHOW SEQUENCE 88
SET
SET ALIAS 190
SET BREAK (type 1) 61
SET BREAK (type 2) 64
SET BREAK (type 3) 65, 66
SET COVERAGE 112
SET DATABREAK (type 1) 72
SET DATABREAK (type 2) 74
SET DELAY 95
SET EVENT 80
SET GO 46
SET INTERRUPT 32
SET LOGGING 200
SET MAP (type 1) 16
SET MAP (type 2) 18
SET MEMORY 150
SET MODE 6
SET MULTITRACE 99
SET OUTPORT 28
SET OUTPUT 208
SET PATH 180
SET PERFORMANCE 107
SET RADIX 11

SET RAMMONITOR....137
SET REGISTER....157
SET SAMPLING....120
SET SEQUENCE....86
SET SOURCE....13
SET STEP....51
SET TIMERSCALE....9
SET TRACE....126
SET VARIABLE....193
SET VECTOR....35
SET WATCH....40

SHOW
SHOW ALIAS....191
SHOW BREAK....67
SHOW CALLS....122
SHOW COVERAGE....113
SHOW DATABREAK....76
SHOW DELAY....96
SHOW EVENT....82
SHOW GO....47
SHOW INPORT....26
SHOW INTERRUPT....33
SHOW LOGGING....202
SHOW MAP....19
SHOW MEMORY....152
SHOW MODE....8
SHOW MULTITRACE....100
SHOW OBJECT....212
SHOW OUTPORT....30
SHOW OUTPUT....209
SHOW PATH....181
SHOW PERFORMANCE....109
SHOW RADIX....12
SHOW RAMMONITOR....139
SHOW REGISTER....158
SHOW SCOPE....182
SHOW SEQUENCE....88
SHOW SOURCE....14
SHOW STATUS....57
SHOW STEP....53
SHOW SYSTEM....15
SHOW TIMER....123
SHOW TIMERSCALE....10
SHOW TRACE (type 1)....127
SHOW TRACE (type 2)....129, 130
SHOW VARIABLE....195
SHOW VECTOR....36
SHOW VERIFYMODE....23
SHOW WATCHDOG....39

SOURCE
SET SOURCE....13
SHOW SOURCE....14

STATUS
SHOW STATUS....57

STEP
SET STEP....51
SHOW STEP....53
STEP....49

SYSTEM
SHOW SYSTEM....15

T

TIMER
CLEAR TIMER....125
SHOW TIMER....123

TIMERSCALE
SET TIMERSCALE....9
SHOW TIMERSCALE....10

TRACE
CLEAR TRACE....132
DISABLE TRACE....134
ENABLE TRACE....133
SEARCH TRACE....135
SET TRACE....126
SHOW TRACE (type 2)....129, 130

U

UP
UP....183

V

VARIABLE
CANCEL VARIABLE....197
SET VARIABLE....193
SHOW VARIABLE....195

VECTOR
SET VECTOR....35
SHOW VECTOR....36

VERIFY
VERIFY....166

VERIFYMODE
DISABLE VERIFYMODE....22
ENABLE VERIFYMODE....21
SHOW VERIFYMODE....23

W

WATCH
CANCEL WATCH....42
SET WATCH....40

WATCHDOG
DISABLE WATCHDOG....38
ENABLE WATCHDOG....37
SHOW WATCHDOG....39

WHILE
WHILE....219

CM25-00323-5E

FUJITSU SEMICONDUCTOR • CONTROLLER MANUAL

F^2MC-8L/8FX FAMILY

SOFTUNE™ Workbench

COMMAND REFERENCE MANUAL

July 2011 the fifrh edition

Published FUJITSU SEMICONDUCTOR LIMITED

Edited Sales Promotion Dept.